

**SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN  
OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA BERBASIS *WEB*  
MENGUNAKAN *FRAMEWORK* LARAVEL**

**(Skripsi)**

**Oleh**

**ALFIAN KAMESWARA**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2021**

## **ABSTRACT**

### **STAFFING INFORMATION SYSTEMS OF YOUTH AND SPORTS SERVICES (DISPORA) PROVINCE OF JAKARTA WEB-BASED USING LARAVEL FRAMEWORK**

**By**

**ALFIAN KAMESWARA**

Every agency, office or university certainly has employee data collection activities and information in its business processes. Staffing information system at the Department of Youth and Sports, one of which is the management of staffing data. The information system and employee data management is very important and becomes a reference in every employee's data collection. At present the compilation of staffing data is still done manually at the agency, will be hampered and become a problem if there is no system that handles these activities, one of which is the search for data and storage space. The staffing information system has been developed to assist in the preparation and recording of staffing activities. This information system was developed using the waterfall method and laravel framework. The conclusion of this study is that the staffing information system was successfully built and can be used to assist in the preparation and search of employee data.

**Keywords:** Staffing Information System, Laravel Framework, Waterfall Methods.

## **ABSTRAK**

### **SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA BERBASIS *WEB* MENGUNAKAN *FRAMEWORK* LARAVEL**

**Oleh**

**ALFIAN KAMESWARA**

Setiap instansi, kantor ataupun perguruan tinggi tentu memiliki kegiatan pendataan pegawai serta informasi dalam proses bisnisnya. Sistem informasi kepegawaian pada Dinas Pemuda dan Olahraga salah satunya adalah pengelolaan data kepegawaian. Sistem informasi serta pengelolaan data pegawai tersebut sangat penting dan menjadi acuan dalam setiap pendataan pegawai. Pada saat ini penyusunan data kepegawaian masih dilakukan secara manual pada instansi tersebut, akan menjadi terhambat dan menjadi masalah apabila tidak ada sistem yang menangani aktifitas tersebut, salah satunya yaitu pencarian data serta ruang penyimpanannya. Sistem informasi kepegawaian telah dikembangkan untuk membantu kegiatan penyusunan dan perekapan data kepegawaian. Sistem informasi ini dikembangkan dengan menggunakan metode *waterfall* dan *laravel framework*. Kesimpulan dari penelitian ini adalah sistem informasi kepegawaian berhasil dibangun dan dapat digunakan untuk membantu kegiatan penyusunan serta pencarian data kepegawaian.

Kata Kunci: Sistem Informasi Kepegawaian, *Laravel Framework*, Metode *Waterfall*.

**SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN  
OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA BERBASIS *WEB*  
MENGUNAKAN *FRAMEWORK* LARAVEL**

**Oleh**

**ALFIAN KAMESWARA**

**Skripsi**

**Sebagai Salah Satu Syarat untuk Memperoleh Gelar  
SARJANA KOMPUTER**

**Pada**

**Jurusan Ilmu Komputer  
Fakultas Matematika dan Ilmu Pengetahuan Alam**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2021**

Judul Skripsi : **SISTEM INFORMASI KEPEGAWAIAN  
DINAS PEMUDA DAN OLAHRAGA  
(DISPORA) PROVINSI DKI JAKARTA  
BERBASIS *WEB* MENGGUNAKAN  
*FRAMEWORK* LARAVEL**

Nama Mahasiswa : **Alfian Kameswara**

Nomor Pokok Mahasiswa : **1417051007**

Jurusan : **Ilmu Komputer**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**

**MENYETUJUI**

**1. Komisi Pembimbing**

**Ir. Machudor Yusman, M.Kom.**

**NIP 19570330 198603 1 003**

**2. Ketua Jurusan Ilmu Komputer**

**Didik Kurniawan, S.Si., M.T.**

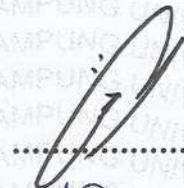
**NIP 19800419 200501 1 004**

## MENGESAHKAN

### 1. Tim Penguji

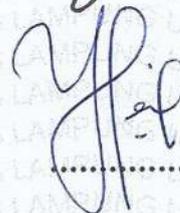
Ketua

: **Ir. Machudor Yusman, M.Kom.**



Penguji I

Bukan Pembimbing : **Anie Rose Irawati, S.T., M.Cs.**



Penguji II

Bukan Pembimbing : **Dr. Rer. Nat. Akmal Junaidi, S.Si., M.Sc.**



### 2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



**Dr. Eng. Supto Dwi Yuwono, M.T.**

NIP 19740705 200003 1 001



Tanggal Lulus Ujian Skripsi : **22 Juli 2021**

## PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul “**Sistem Informasi Kepegawaian Dinas Pemuda Dan Olahraga (DISPORA) Provinsi DKI Jakarta Berbasis Web Menggunakan Framework Laravel**” merupakan karya saya sendiri bukan hasil karya orang lain. Semua tulisan yang tertuang di skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung.

Apabila terbukti di kemudian hari bahwa skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 22 Juli 2021



Alfian Kameswara

NPM. 1417051007

## RIWAYAT HIDUP



Penulis dilahirkan di Provinsi DKI Jakarta pada tanggal 15 Februari 1996, sebagai anak pertama dari dua bersaudara dengan ayah bernama Darmawan dan ibu bernama Nedrawati. Penulis memiliki satu adik perempuan yang bernama Ulfah Dwiki Regina.

Penulis menyelesaikan pendidikan Taman Kanak-Kanak (TK) pada tahun 2002 di TKS Hang Tuah 6 Jakarta, Sekolah Dasar Swasta (SDS) di SDS Hang Tuah 7 Jakarta pada tahun 2008, Sekolah Menengah Pertama (SMP) di SMP Negeri 27 Bandar Lampung pada tahun 2011, Sekolah Menengah Kejuruan (SMK) di SMK Negeri 1 Bandar Lampung pada tahun 2014.

Pada tahun 2014, penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung. Selama menjadi mahasiswa penulis aktif mengikuti beberapa kegiatan, antara lain :

1. Anggota Abacus Himakom 2014-2015.
2. Pada bulan Januari 2015 penulis mengikuti Karya Wisata Ilmiah di Desa Sidokaton, Kecamatan Gisting, Kabupaten Tanggamus.
3. Anggota Biro Kewirausahaan Himakom 2015-2016.
4. Sekretaris Biro Kesekretariatan Himakom 2016-2017.
5. Melaksanakan Kerja Praktik di Dinas Pemuda Dan Olahraga Provinsi DKI Jakarta (Bagian Kepegawaian) Provinsi DKI Jakarta tanggal 16 Januari - 24 Februari 2017.
6. Pada bulan Juli-September 2017, penulis mengikuti Kuliah Kerja Nyata selama 40 hari di Desa Pagar Alam, Kecamatan Ulu Belu, Kabupaten Tanggamus.

## **PERSEMBAHAN**

*Puji dan syukur saya panjatkan kepada Allah Swt, atas segala berkat, rahmat dan hidayahNya sehingga skripsi ini dapat terselesaikan.*

*Kupersembahkan karya kecilku ini untuk:*

*Papa dan Mama yang sangat kucintai yang telah membesarkan, mendidik, menjaga, melindungi, memberikan motivasi, dukungan dan doa yang tulus tiada henti demi kesuksesan anak-anaknya.*

*Adik ku Ulfah Dwiki Regina dan keluarga besar yang selalu mendoakan dan mendukung.*

*Sahabat dan teman teman seperjuangan, terima kasih untuk segala cerita yang pernah terukir baik suka maupun duka, terima kasih telah dan selalu menjadi bagian dari hidup ini.*

*Setiap orang yang bertanya “Kapan Wisuda?” terima kasih atas doa, motivasi dan dukungannya.*

*Keluarga Besar Ilmu Komputer 2014 dan Almamater yang Kubanggakan  
UNIVERSITAS LAMPUNG.*

## **MOTTO**

*“Wahai orang-orang yang beriman, jadikanlah sabar dan shalat sebagai penolongmu, sesungguhnya Allah beserta orang-orang yang sabar”*

*[Q.S Al-Baqarah : 153]*

*“Maka sesungguhnya bersama kesulitan itu ada kemudahan. Sesungguhnya bersama kesulitan itu ada kemudahan”*

*[Q.S Al-Alinsyirah : 5-6]*

*“Berdoalah (mintalah) kepadaKu (Allah SWT), pastilah aku kabulkan untukmu.”*

*[Q.S Al-Mukmin : 60]*

*“Think Before You Act”*

*[Alfian Kameswara]*

## SANWACANA

Alhamdulillah, puji syukur kehadiran Allah SWT, atas berkat rahmat, hidayah, dan kesehatan yang diberikan sehingga penulis dapat menyelesaikan skripsi yang berjudul “Sistem Informasi Kepegawaian Dinas Pemuda Dan Olahraga (DISPORA) Provinsi DKI Jakarta Berbasis *Web* Menggunakan *Framework* Laravel”.

Dalam penyusunan skripsi ini, penulis mendapatkan bantuan, dukungan dan dorongan dari berbagai pihak. Oleh karena itu, terima kasih penulis sampaikan kepada semua pihak yang telah membantu dan berperan besar dalam menyusun skripsi ini, antara lain:

1. Kedua orang tua tercinta, Mama dan Papa yang selalu mendoakan kesuksesan penulis, memberikan dukungan moril maupun materil, membimbing, memberikan motivasi, kasih sayang yang tak terhingga dan menghargai setiap proses penulis selama ini.
2. Adikku Ulfah Dwiki Regina S.Pd.I yang selalu mendoakan, mendukung dan menghibur penulis.
3. Bapak Ir. Machudor Yusman M.Kom. sebagai Pembimbing Utama yang telah membimbing serta memberikan ide, kritik dan saran selama masa perkuliahan dan penyusunan skripsi sehingga penulis bisa sampai ditahap ini.
4. Ibu Anie Rose Irawati S.T., M.Cs. sebagai Pembahas I yang telah memberikan kritik dan saran selama masa perkuliahan dan penyusunan skripsi, sehingga penulis bisa sampai ditahap ini.
5. Bapak Dr. Rer. Nat. Akmal Junaidi S.Si., M.Sc. sebagai Pembahas II yang telah memberikan banyak masukan, ide, kritik serta saran yang sangat bermanfaat dalam proses penyelesaian skripsi ini.

6. Bapak Dr. Eng. Admi Syarif sebagai Pembimbing Akademik yang telah membimbing, memotivasi serta memberikan ide, kritik dan saran selama masa perkuliahan.
7. Bapak Dr. Suropto Dwi Yuwono M.T. sebagai Dekan FMIPA Universitas Lampung.
8. Bapak Didik Kurniawan S.Si., M.T. sebagai Ketua Jurusan Ilmu Komputer FMIPA Universitas Lampung.
9. Ibu Astria Hijriani S.Kom., M.Kom. sebagai Sekretaris Jurusan Ilmu Komputer FMIPA Universitas Lampung.
10. Seluruh Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah memberikan ilmu dan pengalaman hidup selama masa perkuliahan.
11. Ibu Ade Nora Maela dan Ibu Lusiana yang telah membantu segala urusan administrasi penulis di Jurusan Ilmu Komputer FMIPA Universitas Lampung.
12. Mas Ardi Nofalian yang selalu membantu menyediakan dan memberikan tempat untuk melaksanakan seminar serta mengerjakan skripsi.
13. Andri Desmana Putra S.Kom., Faiz Azmi Rekatama S.Kom, yang menjadi tempat *sharing* penulis ketika bertemu dengan *error* dan hal-hal yang tidak diketahui oleh penulis.
14. Faqiha Rasyid F S.Sos., Hendri Mustakim S.Si., Herlina S.Sc., Hollyati Subhi Gurnita S.E., rekan Kuliah Kerja Nyata yang selalu menemani, mendoakan, mendukung dan menghibur penulis serta menjadi tempat berbagi suka dan duka. Sukses dan sehat selalu.
15. Kawan-kawan BUJANG : Andi Marta Dinata S.Kom., Sigit Prabowo S.Kom., Aditya Riyaldie S.Kom., M. Iqbal Nugraha S.Kom., M. Juandika Riski S.Kom., M. Haekal S.Kom., Nur M. Syaifullah S.Kom., Gregorius F S.Kom., Ragananda Rinaldi S.Kom., Rama Bayu N S.Kom., Robi M.P S.Kom., Thomi Fadholi S.Kom., Tungki Adi P S.Kom., Widyan Hasbi S.Kom., M. Doni Syatria S.Kom., yang telah memberikan bantuan, doa, dukungan dan menghibur penulis.

16. M. Berlian Alfaridz rekan satu kerja praktik selama di Dinas Pemuda Dan Olahraga (DISPORA) Provinsi DKI Jakarta serta rekan selama masa perkuliahan dan skripsi.
17. Sahabat Ciangsana : Fachrul Hafiz A, Tedi Ahmad S, Kurnia Aji W, Satrio W, M. Ade Suryana, Adi Tri Laksono P, A. Tri Rahmadini, Bimo Wahyudianto, M. W. Fazrin, Fadhlan Pangestu, Tugas Ardiyana S, Rahul, Andri, yang telah memberikan doa, dukungan serta menghibur penulis dalam menyelesaikan skripsi ini.
18. Keluarga Besar Ilmu Komputer 2014 yang telah berjuang bersama dan memberikan kenangan dalam menjalankan studi di Jurusan Ilmu Komputer Universitas Lampung.
19. Almamater tercinta Universitas Lampung yang telah memberikan penulis kesempatan untuk menempuh pendidikan sarjana dengan baik.

Penulis

**Alfian Kameswara**

## DAFTAR ISI

	Halaman
<b>DAFTAR ISI</b> .....	<b>i</b>
<b>DAFTAR GAMBAR</b> .....	<b>iii</b>
<b>DAFTAR TABEL</b> .....	<b>vii</b>
<b>DAFTAR KODE</b> .....	<b>ix</b>
<b>I. PENDAHULUAN</b>	
1.1 Latar Belakang .....	1
1.2 Identifikasi Masalah .....	4
1.3 Rumusan Masalah .....	4
1.4 Batasan Masalah .....	4
1.5 Manfaat Penelitian .....	5
1.6 Tujuan Penelitian .....	5
<b>II. TINJAUAN PUSAKA</b>	
2.1 Definisi Komputer .....	6
2.2 Sistem Informasi .....	6
2.2.1 Definisi Sistem .....	6
2.2.2 Definisi Informasi.....	7
2.2.3 Kesimpulan Sistem Informasi .....	7
2.3 Pengertian Pegawai .....	8
2.4 Metode Pengembangan Sistem .....	9
2.5 Pengertian <i>Unified Modelling Language (UML)</i> .....	9
2.6 Komponen - Komponen <i>Unified Modelling Language (UML)</i> .....	10
2.6.1 <i>Use Case Diagram</i> .....	10
2.6.2 <i>Activity Diagram</i> .....	12
2.6.3 <i>Sequence Diagram</i> .....	13
2.6.4 <i>Class Diagram</i> .....	14
2.7 Pengertian <i>Website</i> .....	15
2.8 <i>Software</i> Pendukung Pengembangan Sistem Kepegawaian .....	15
2.8.1 <i>PHP (Hypertext Preprocessor)</i> .....	15
2.8.2 <i>Laravel</i> .....	16
2.8.3 Definisi <i>MVC (Model View Controller)</i> .....	19
2.8.4 <i>Framework</i> .....	19
2.8.5 <i>MySQL</i> .....	20
2.8.6 <i>XAMPP</i> .....	21
2.8.7 <i>Dreamweaver</i> .....	22
2.8.8 <i>Database</i> .....	22

2.8.9 <i>HTML (Hypertext Markup Language)</i> .....	22
2.8.10 Definisi <i>Black Box Testing</i> .....	23

### III. METODE PENELITIAN

3.1 Waktu dan Tempat Penelitian .....	25
3.2 Alat Pendukung .....	25
3.3 Tahapan Penelitian .....	26
3.4 Pengajuan Solusi Alternatif .....	28
3.4.1 <i>Use Case Diagram</i> .....	29
3.4.2 <i>Activity Diagram</i> .....	40
3.4.3 <i>Sequence Diagram</i> .....	61
3.4.4 <i>Class Diagram</i> .....	88
3.4.5 Rancangan <i>Input</i> untuk Akses <i>Admin</i> .....	96
3.4.6 Rancangan <i>Input</i> untuk Akses Pegawai .....	109
3.4.7 Rancangan untuk Akses Kepala Dinas .....	115

### IV. HASIL DAN PEMBAHASAN

4.1 Hasil .....	131
4.2 Implementasi Sistem .....	132
4.3 Tampilan Sistem Kepegawaian DISPORA Berbasis <i>Web</i> .....	137
4.3.1 Tampilan pada pengguna tidak terautentikasi ( <i>Guest</i> ) .....	138
4.3.2 Tampilan Pada Admin .....	140
4.3.3 Tampilan Pada Pegawai .....	200
4.3.4 Tampilan Pada Kadis .....	211
4.4 Pengujian Sistem .....	219
4.5 Evaluasi .....	228

### V. KESIMPULAN DAN SARAN

5.1 Kesimpulan .....	232
5.2 Saran .....	233

### DAFTAR PUSTAKA

### LAMPIRAN

## DAFTAR GAMBAR

Gambar	Halaman
Gambar 3.1 Metode <i>Waterfall</i> .....	26
Gambar 3.2 <i>Use Case Diagram</i> Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta.....	30
Gambar 3.3 <i>Activity Diagram Login User</i> .....	41
Gambar 3.4 <i>Activity Diagram</i> Kelola Data Pegawai.....	42
Gambar 3.5 <i>Activity Diagram</i> Kelola Data Cuti Pegawai.....	43
Gambar 3.6 <i>Activity Diagram Input</i> Data Pengajuan Cuti.....	44
Gambar 3.7 <i>Activity Diagram</i> Kelola Data Absensi Pegawai.....	45
Gambar 3.8 <i>Activity Diagram Input</i> Absen.....	46
Gambar 3.9 <i>Activity Diagram</i> Kelola Data Pangkat Pegawai.....	47
Gambar 3.10 <i>Activity Diagram</i> Kelola Data Pendidikan Terakhir .....	48
Gambar 3.11 <i>Activity Diagram</i> Kelola Data Mutasi Pegawai.....	49
Gambar 3.12 <i>Activity Diagram</i> Kelola Data Pensiun Pegawai .....	50
Gambar 3.13 <i>Activity Diagram</i> Mengganti <i>Password</i> .....	51
Gambar 3.14 <i>Activity Diagram</i> <i>Monitoring</i> Data.....	52
Gambar 3.15 <i>Activity Diagram</i> Cetak Laporan.....	53
Gambar 3.16 <i>Activity Diagram</i> Kelola Data Berkas .....	54
Gambar 3.17 <i>Activity Diagram Input</i> Berkas.....	55
Gambar 3.18 <i>Activity Diagram Input</i> Data Pegawai.....	56
Gambar 3.19 <i>Activity Diagram Input</i> Data Pensiun.....	57
Gambar 3.20 <i>Activity Diagram Input</i> Data Mutasi Pegawai.....	58
Gambar 3.21 <i>Activity Diagram</i> <i>Monitoring</i> Data Kepala Dinas .....	59
Gambar 3.22 <i>Activity Diagram</i> <i>Verifikasi</i> Berkas.....	60
Gambar 3.23 <i>Sequence Diagram</i> <i>Login User</i> .....	61
Gambar 3.24 <i>Sequence Diagram</i> Kelola Data Pegawai.....	62

Gambar 3.25 <i>Sequence Diagram</i> Kelola Data Cuti Pegawai.....	64
Gambar 3.26 <i>Sequence Diagram Input</i> Data Pengajuan Cuti.....	66
Gambar 3.27 <i>Sequence Diagram</i> Kelola Data Absensi Pegawai.....	67
Gambar 3.28 <i>Sequence Diagram Input</i> Absen.....	68
Gambar 3.29 <i>Sequence Diagram</i> Kelola Data Pangkat Pegawai .....	69
Gambar 3.30 <i>Sequence Diagram</i> Kelola Data Pendidikan Terakhir .....	71
Gambar 3.31 <i>Sequence Diagram</i> Kelola Data Mutasi Pegawai .....	73
Gambar 3.32 <i>Sequence Diagram</i> Kelola Data Pensiun Pegawai.....	75
Gambar 3.33 <i>Sequence Diagram</i> Mengganti <i>Password</i> .....	77
Gambar 3.34 <i>Sequence Diagram</i> <i>Monitoring</i> Data.....	78
Gambar 3.35 <i>Sequence Diagram</i> Cetak Laporan.....	79
Gambar 3.36 <i>Sequence Diagram</i> Kelola Data Foto.....	80
Gambar 3.37 <i>Sequence Diagram Input</i> Foto.....	82
Gambar 3.38 <i>Sequence Diagram Input</i> Data Pegawai.....	83
Gambar 3.39 <i>Sequence Diagram Input</i> Data Pensiun.....	84
Gambar 3.40 <i>Sequence Diagram Input</i> Data Mutasi Pegawai.....	85
Gambar 3.41 <i>Sequence Diagram</i> <i>Monitoring</i> Data Kadis .....	86
Gambar 3.42 <i>Sequence Diagram</i> <i>Verifikasi</i> Berkas.....	87
Gambar 3.43 <i>Class Diagram</i> Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta .....	88
Gambar 3.44 Desain <i>Interface</i> <i>Login</i> User Dan Admin .....	97
Gambar 3.45 Desain <i>Interface</i> Kelola Data Pegawai.....	98
Gambar 3.46 Desain <i>Interface</i> Kelola Data Cuti Pegawai.....	99
Gambar 3.47 Desain <i>Interface</i> Kelola Data Absensi Pegawai.....	100
Gambar 3.48 Desain <i>Interface</i> Kelola Data Pangkat Pegawai.....	101
Gambar 3.49 Desain <i>Interface</i> Kelola Data Pendidikan Terakhir .....	102
Gambar 3.50 Desain <i>Interface</i> Kelola Data Mutasi Pegawai.....	103
Gambar 3.51 Desain <i>Interface</i> Kelola Data Pensiun Pegawai .....	104
Gambar 3.52 Desain <i>Interface</i> Mengganti <i>Password</i> .....	105
Gambar 3.53 Desain <i>Interface</i> <i>Monitoring</i> Data.....	106
Gambar 3.54 Desain <i>Interface</i> Cetak Laporan.....	107
Gambar 3.55 Desain <i>Interface</i> Kelola Data Foto.....	108
Gambar 3.56 Desain <i>Interface Input</i> Data Pengajuan Cuti .....	109

Gambar 3.57 Desain <i>Interface Input Absen</i> .....	110
Gambar 3.58 Desain <i>Interface Input Foto</i> .....	111
Gambar 3.59 Desain <i>Interface Input Data Pegawai</i> .....	112
Gambar 3.60 Desain <i>Interface Input Data Pensiun</i> .....	113
Gambar 3.61 Desain <i>Interface Input Data Mutasi Pegawai</i> .....	114
Gambar 3.62 Desain <i>Interface Verifikasi Berkas Pengajuan Cuti</i> .....	115
Gambar 3.63 Desain <i>Interface Verifikasi Berkas Pengajuan Mutasi</i> .....	116
Gambar 3.64 Desain <i>Interface Verifikasi Berkas Pengajuan Pensiun</i> .....	117
Gambar 3.65 Desain <i>Interface Monitoring Data Dan Grafik Data</i> .....	118
Gambar 4.1 Halaman <i>Utama</i> .....	140
Gambar 4.2 Halaman <i>Login</i> .....	141
Gambar 4.3 Halaman <i>Menu Data Pegawai</i> .....	144
Gambar 4.4 Halaman <i>Input Data Pegawai</i> .....	145
Gambar 4.5 Halaman <i>Lihat Data Pegawai</i> .....	146
Gambar 4.6 Halaman <i>Edit Data Pegawai</i> .....	147
Gambar 4.7 Halaman <i>Menghapus Data Pegawai</i> .....	149
Gambar 4.8 Halaman <i>Reset Password Pegawai</i> .....	150
Gambar 4.9 Halaman <i>Menu Data Kadis</i> .....	151
Gambar 4.10 Halaman <i>Edit Data Kadis</i> .....	152
Gambar 4.11 Halaman <i>Menu Data Pangkat</i> .....	154
Gambar 4.12 Halaman <i>Tambah Data Pangkat</i> .....	155
Gambar 4.13 Halaman <i>Edit Data Pangkat</i> .....	157
Gambar 4.14 Halaman <i>Menu Data Cuti</i> .....	158
Gambar 4.15 Halaman <i>Lihat Data Cuti</i> .....	159
Gambar 4.16 Halaman <i>Menu Data Absen</i> .....	161
Gambar 4.17 Halaman <i>Menu Data Mutasi</i> .....	162
Gambar 4.18 Halaman <i>Input Data Mutasi</i> .....	163
Gambar 4.19 Halaman <i>Menu Data Pensiun</i> .....	165
Gambar 4.20 Halaman <i>Tambah Data Pensiun</i> .....	166
Gambar 4.21 Halaman <i>Lihat Data Pensiun</i> .....	167
Gambar 4.22 Halaman <i>Menu Laporan Pegawai</i> .....	169
Gambar 4.23 Halaman <i>Menu Laporan Absensi</i> .....	171

Gambar 4.24 Halaman <i>Menu</i> Cuti .....	173
Gambar 4.25 Halaman Pengajuan Cuti .....	174
Gambar 4.26 Halaman <i>Menu</i> Absen .....	175
Gambar 4.24 Halaman <i>Input</i> Absen .....	176
Gambar 4.28 Halaman <i>Menu</i> Mutasi .....	178
Gambar 4.29 Halaman Pengajuan Mutasi.....	179
Gambar 4.30 Halaman <i>Menu</i> Pensiun.....	181
Gambar 4.31 Halaman <i>Input</i> Pensiun .....	182
Gambar 4.32 Halaman <i>Menu Verifikasi</i> Cuti.....	184
Gambar 4.33 Halaman <i>Menu Verifikasi</i> Mutasi.....	186
Gambar 4.34 Halaman Aksi <i>Verifikasi</i> Cuti dan Mutasi.....	188
Gambar 4.35 Halaman <i>Menu</i> Laporan Pegawai .....	189

## DAFTAR TABEL

Tabel	Halaman
Tabel 2.1 Simbol <i>Use Case Diagram</i> .....	11
Tabel 2.2 Simbol <i>Activity Diagram</i> .....	12
Tabel 2.3 Simbol <i>Sequence Diagram</i> .....	13
Tabel 2.4 Simbol <i>Class Diagram</i> .....	14
Tabel 3.1 Definisi Aktor .....	31
Tabel 3.2 Definisi <i>Use Case Diagram</i> .....	31
Tabel 3.3 Skenario <i>Use Case Login</i> sebagai Pegawai .....	33
Tabel 3.4 Skenario <i>Use Case Kelola Data Pegawai</i> .....	33
Tabel 3.5 Skenario <i>Use Case Kelola Data Cuti Pegawai</i> .....	34
Tabel 3.6 Skenario <i>Use Case Input Data Pengajuan Cuti</i> .....	34
Tabel 3.7 Skenario <i>Use Case Login</i> sebagai <i>Admin</i> .....	34
Tabel 3.8 Skenario <i>Use Case Kelola Data Absensi Pegawai</i> .....	35
Tabel 3.9 Skenario <i>Use Case Input Absen</i> .....	35
Tabel 3.10 Skenario <i>Use Case Kelola Data Pangkat Pegawai</i> .....	35
Tabel 3.11 Skenario <i>Use Case Kelola Data Pendidikan Terakhir</i> .....	36
Tabel 3.12 Skenario <i>Use Case Kelola Data Mutasi Pegawai</i> .....	36
Tabel 3.13 Skenario <i>Use Case Kelola Data Pensiun Pegawai</i> .....	37
Tabel 3.14 Skenario <i>Use Case Input Data Pensiun</i> .....	37
Tabel 3.15 Skenario <i>Use Case Cetak Laporan</i> .....	37
Tabel 3.16 Skenario <i>Use Case Monitoring Data</i> .....	38
Tabel 3.17 Skenario <i>Use Case Kelola Data Foto</i> .....	38
Tabel 3.18 Skenario <i>Use Case Input Foto</i> .....	38
Tabel 3.19 Skenario <i>Use Case Input Data Pegawai</i> .....	38
Tabel 3.20 Skenario <i>Use Case Mengganti Password</i> .....	39
Tabel 3.21 Skenario <i>Use Case Kelola Data Pegawai</i> .....	39

Tabel 3.22 Skenario <i>Use Case Input</i> Data Mutasi Pegawai.....	39
Tabel 3.23 Skenario <i>Use Case Login</i> sebagai Kepala Dinas .....	40
Tabel 3.24 Skenario <i>Use Case Monitoring</i> Data .....	40
Tabel 3.25 Skenario <i>Use Case Verifikasi</i> Berkas .....	40
Tabel 3.26 Kamus data tabel pegawai.....	89
Tabel 3.27 Kamus tabel laporan.....	90
Tabel 3.28 Kamus tabel ganti <i>password</i> .....	90
Tabel 3.29 Kamus data tabel foto .....	91
Tabel 3.30 Kamus data tabel pendidikan terakhir.....	91
Tabel 3.31 Kamus data tabel absen.....	92
Tabel 3.32 Kamus data tabel mutasi .....	93
Tabel 3.33 Kamus data tabel pensiun .....	94
Tabel 3.34 Kamus data tabel pangkat .....	95
Tabel 3.35 Kamus data tabel cuti.....	95
Tabel 3.36 Daftar Pengujian pada <i>Admin</i> .....	119
Tabel 3.37 Daftar Pengujian pada Pegawai .....	129
Tabel 3.38 Daftar Pengujian pada Kadis.....	133
Tabel 4.1 Daftar <i>Model</i> Sistem Informasi Kepegawaian DISPORA.....	137
Tabel 4.2 Daftar <i>Views</i> Sistem Informasi Kepegawaian DISPORA.....	137
Tabel 4.3 Tabel Pengujian Pada <i>Admin</i> .....	192
Tabel 4.4 Tabel Pengujian Pada Pegawai .....	198
Tabel 4.5 Tabel Pengujian Pada Kadis .....	200
Tabel 4.6 Daftar Pertanyaan Pada Kuesioner .....	203
Tabel 4.7 Daftar Pertanyaan Pada Kuesioner .....	204
Tabel 4.8 Kategori Penilaian.....	205

## DAFTAR KODE

Kode	Halaman
Kode 1. Potongan kode program halaman utama .....	140
Kode 2. Potongan kode program <i>button login</i> .....	142
Kode 3. Potongan kode halaman <i>menu</i> data pegawai .....	144
Kode 4. Potongan kode halaman <i>input</i> data pegawai.....	145
Kode 5. Potongan kode halaman lihat data pegawai.....	147
Kode 6. Potongan kode halaman <i>edit</i> data pegawai.....	148
Kode 7. Potongan kode halaman menghapus data pegawai.....	149
Kode 8. Potongan kode halaman <i>reset password</i> pegawai .....	150
Kode 9. Potongan kode halaman <i>menu</i> data kadis .....	152
Kode 10. Potongan kode halaman <i>edit</i> data kadis.....	153
Kode 11. Potongan kode halaman <i>menu</i> data pangkat .....	154
Kode 12. Potongan kode halaman tambah data pangkat.....	156
Kode 13. Potongan kode halaman <i>edit</i> data pangkat pegawai .....	157
Kode 14. Potongan kode halaman <i>menu</i> data cuti.....	159
Kode 15. Potongan kode halaman lihat data cuti .....	160
Kode 16. Potongan kode halaman <i>menu</i> data absen .....	161
Kode 17. Potongan kode halaman <i>menu</i> data mutasi.....	163
Kode 18. Potongan kode halaman <i>input</i> data mutasi .....	164
Kode 19. Potongan kode halaman <i>menu</i> data pensiun .....	166
Kode 20. Potongan kode halaman tambah data pensiun.....	167
Kode 21. Potongan kode halaman lihat data pensiun.....	168
Kode 22. Potongan kode halaman laporan pegawai .....	170
Kode 23. Potongan kode halaman <i>menu</i> laporan absensi .....	171
Kode 24. Potongan kode halaman <i>menu</i> cuti .....	173
Kode 25. Potongan kode halaman pengajuan cuti .....	174

Kode 26. Potongan kode halaman <i>menu</i> absen.....	176
Kode 27. Potongan kode halaman pengajuan absen .....	177
Kode 28. Potongan kode halaman <i>menu</i> mutasi .....	178
Kode 29. Potongan kode halaman pengajuan mutasi.....	180
Kode 30 Potongan kode halaman <i>menu</i> pensiun.....	181
Kode 31. Potongan kode halaman pengajuan pensiun.....	182
Kode 32. Potongan kode halaman <i>menu verifikasi</i> cuti .....	185
Kode 33. Potongan kode halaman <i>menu verifikasi</i> mutasi .....	187
Kode 34. Potongan kode halaman <i>menu</i> laporan pegawai.....	190

# I. PENDAHULUAN

## 1.1 Latar Belakang

Setiap instansi pemerintahan tentu membutuhkan sistem informasi dalam menjalankan aktivitas kerjanya, untuk menghasilkan kinerja yang lebih teratur dan terarah dengan waktu yang lebih cepat. Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta merupakan salah satu instansi pemerintahan yang telah memakai sistem informasi berupa pemakaian perangkat komputer dalam menjalankan aktivitas kerjanya.

Pada saat ini, pengelolaan data kepegawaian di Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta masih dilakukan dengan cara pembukuan maupun memasukkan data kedalam *microsoft excel*, dengan demikian untuk suatu instansi pemerintahan kurang tepat jika menggunakan aplikasi tersebut. Maksud dari sistem informasi adalah perangkat lunak yang memberikan informasi untuk memudahkan pencarian data pegawai secara akurat dan tepat. Daftar semua data pegawai yang ada dalam instansi Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta ini dari setiap wilayah ataupun percabangannya, data disimpan dalam urutan abjad.

Sedangkan kepegawaian adalah sebuah bidang yang terdapat dalam suatu instansi pemerintah maupun perusahaan. Pengelolaan data kepegawaian yang masih

dilakukan dengan manual dapat mempersulit pegawai saat melakukan pendataan dan pembuatan laporan, sehingga dapat memakan waktu yang lebih lama. Belum adanya sistem informasi untuk mengelola data kepegawaian yang mengakibatkan seringnya terjadi kesalahan dalam melakukan pendataan pegawai di Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta.

Menurut Alfian Kameswara pada laporan kegiatan Praktik Kerja Lapangan yang berjudul Rancangan Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta menyatakan bahwa Sistem Informasi Kepegawaian yang dirancang dapat diimplementasikan menjadi sebuah sistem yang dapat digunakan bagi pegawai Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta, sebagai suatu sarana untuk mempermudah pekerjaan pegawai dalam menyimpan data pegawai serta meningkatkan kinerja pegawai.

Selama ini, dalam proses pengolahan data di kantor Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta masih dilakukan dengan cara menulis didalam buku besar dan pengarsipan disimpan dalam bentuk *hardcopy*, lalu diletakkan dalam map besar. Oleh karena itu sering terjadi kekeliruan dalam proses penyimpanan data pegawai. Suatu pemrograman yang memberikan kemudahan dalam mengakses berbagai informasi dan memiliki keamanan dalam menyimpan data diantaranya adalah *Web* yakni, merupakan sebuah penyebaran informasi melalui internet, dengan adanya *Web* dapat memberikan kemudahan dalam mengakses informasi serta dapat berinteraksi, dapat menyimpan *file* melalui *web*, memiliki keamanan dan kenyamanan. *Web* juga merupakan media dengan jejaring terluas di dunia, tidak terbatas oleh waktu dan tidak tersekat oleh batas wilayah.

Oleh karena itu dengan adanya masalah tersebut maka perlu untuk dirancang serta dibangunnya sistem informasi kepegawaian di Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta. Dalam pembangunan sistem informasi kepegawaian ini yang berbasis *web* menggunakan *framework laravel*, sebagai pengajuan skripsi untuk membantu pegawai DISPORA dalam pengolahan data pegawai.

Sistem informasi kepegawaian ini dirancang untuk memudahkan pekerjaan pegawai dalam pengarsipan data, masih terdapat kekurangan sistem yang digunakan dalam hal pengumpulan data pegawai, maka diperlukannya sistem penyimpanan data yang dapat menyimpan secara rapih, mudah dicari kembali pada saat data diperlukan dengan keamanan lebih akurat. Dizaman yang semakin canggih ini sebuah instansi pemerintahan tentu perlu untuk selalu memperbaharui serta mengetahui tentang bagaimana perkembangan teknologi yang dipakai saat ini, sebagai salah satu penunjang perkembangan dari instansi tersebut. Desain *interface* dirancang atas saran dan masukan yang diberikan oleh pegawai yang bertugas memegang data kepegawaian.

Pendataan pegawai saat ini menjadi salah satu masalah nantinya jika tidak ada suatu sistem informasi yang menangani aktifitas perekapan data pegawai tersebut. Salah satu masalah dari pendataan yang akan muncul yaitu di dalam pencarian data pegawai serta ruang penyimpanannya. Oleh karena itu, dibutuhkan suatu sistem informasi untuk menangani aktifitas pendataan tersebut, karena pada dasarnya sistem informasi merupakan faktor yang berperan penting dan terkait dalam menunjang jalannya proses bisnis. *Laravel* merupakan suatu *framework*

PHP *open source* yang dibangun dengan menggunakan konsep MVC (*Model- View- Controller*) yang memiliki tiga kelebihan utama yaitu *expressif*, *simple* dan *accessible* (Aminudin, 2015).

Hal tersebut yang menjadi dasar untuk mengembangkan sistem informasi kepegawaian sebagai sarana untuk membantu kegiatan penyusunan, pencarian data kepegawaian pada DISPORA Provinsi DKI Jakarta.

## **1.2 Identifikasi Masalah**

Berdasarkan latar belakang di atas, maka dapat diidentifikasi masalah-masalah sebagai berikut :

- a. Pegawai masih melakukan penyimpan data dalam bentuk berkas (*hardcopy*),
- b. Dalam proses pengarsipan data sering terjadinya kesalahan memasukkan data.

## **1.3 Rumusan Masalah**

Rumusan masalah dalam pembuatan Skripsi ini yaitu bagaimana merancang dan membangun Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta menggunakan *Framework* Laravel, untuk membantu serta mempermudah pegawai dalam pengarsipan data pegawai, supaya data dapat tersimpan dengan lebih baik dan aman.

## **1.4 Batasan Masalah**

Batasan masalah dalam penelitian Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta adalah sebagai berikut :

- a. Data yang digunakan adalah data Kepegawaian DISPORA Provinsi DKI Jakarta,
- b. Sistem ini berbasis *Web* menggunakan *Framework* Laravel.

### **1.5 Manfaat Penelitian**

Manfaat yang diharapkan dari pembuatan Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta adalah :

- a. Mempermudah dalam pencarian informasi data seorang pegawai,
- b. Data dapat diunduh dengan menggunakan format *Portable Document File* (PDF) atau format *excel* (.xls),
- c. Mendapatkan informasi tentang keadaan pegawai dengan cepat dan akurat,
- d. Meningkatkan keamanan serta menjaga data yang tersimpan pada *database*,
- e. Pencarian data pegawai dapat lebih mudah, lebih cepat, lebih akurat dan lebih lengkap serta menghemat banyak tenaga dan waktu,
- f. Pencetakan laporan dapat dilakukan setiap saat.

### **1.6 Tujuan Penelitian**

Tujuan dari pembuatan Sistem Informasi Kepegawaian ini adalah membangun sebuah Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta berbasis *Web* dengan *Framework* Laravel.

## **II. TINJAUAN PUSAKA**

### **2.1 Definisi Komputer**

Kata komputer berasal dari bahasa Latin yaitu *computare* yang artinya menghitung. secara definisi komputer diterjemahkan sebagai sekumpulan alat elektronik yang saling bekerjasama, serta dapat menerima data masukan (*input*) dan memberikan informasi (Yuhefizer, 2003). Kata *computer* semula dipergunakan untuk menggambarkan orang yang pekerjaannya melakukan perhitungan aritmatika, dengan atau tanpa alat bantu.

### **2.2 Sistem Informasi**

Uraian tentang definisi sistem informasi berisi pengertian sistem, pengertian informasi dan pengertian sistem informasi.

#### **2.2.1 Definisi Sistem**

Sistem berasal dari bahasa Latin *systema* dan bahasa Yunani *sustema* adalah suatu kegunaan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi. Menurut Jogiyanto Hartono (2001) terdapat dua susunan kelompok yang didalamnya mendefinisikan suatu sistem dan kegiatan-kegiatan yang menekan suatu sistem yang melalui jaringan yang saling berhubungan untuk untuk menyelesaikan suatu tujuan tertentu harus menekankan pada suatu sistem atau elemen-elemen yang dapat

mempermudah mencapai suatu tujuan tertentu. Sistem juga merupakan sebuah tatanan yang terdiri atas sejumlah komponen fungsional yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses tertentu.

### **2.2.2 Definisi Informasi**

Dalam sejarahnya, informasi berasal dari bahasa Prancis kuno *informacion* pada tahun 1387 yang berarti garis besar, konsep, atau ide. Informasi adalah hasil pengolahan data yang memberikan arti dan manfaat.

Pengertian Informasi menurut Jogiyanto Hartono (2001) adalah bahwa informasi merupakan hasil pengolahan data yang diolah menjadi sedemikian rupa sehingga fungsi dan kegunaannya lebih berguna dan berarti bagi pengguna yang lainnya sebagai dasar pengambilan keputusan.

### **2.2.3 Kesimpulan Sistem Informasi**

Sistem Informasi adalah kumpulan dari subsistem baik untuk fisik maupun nonfisik yang saling berhubungan satu sama lain dan bekerja sama untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berarti dan berguna (Susanto, 2004). Sistem Informasi yaitu sistem yang didalamnya terdapat organisasi yang mempertemukan pengolahan data kebutuhan transaksi harian dan mendukung operasi serta kegiatan strategi dari suatu pihak luar instansi atau organisasi untuk menyediakan data atau laporan-laporan yang dibutuhkan atau diperlukan (Kristanto, 2002).

Sistem Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi,

bersifat manajerial dan kegiatan strategi dari suatu organisasi serta menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Jogiyanto. H, 2001). Dari tiga pengertian sistem informasi diatas, maka dapat dikatakan bahwa sistem informasi adalah kumpulan elemen yang berinteraksi untuk mendapatkan tujuan dalam pengolahan data dan menyediakan informasi untuk membantu dalam pengambilan keputusan manajemen, sehingga memberikan data yang bermanfaat untuk mencapai tujuan dan memberikan informasi yang layak untuk pihak luar sebuah instansi.

### **2.3 Pengertian Pegawai**

Pegawai adalah seseorang yang melakukan kehidupannya dengan bekerja pada kesatuan organisasi, baik kesatuan kerja pemerintah maupun kesatuan kerja swasta (Soedaryono, 2000).

Ada dua pengertian pegawai negeri menurut Undang-Undang Pokok Kepegawaian No. 43 Tahun 1999 Tentang Perubahan UU No. 8 tahun 1974 Tentang Pokok-Pokok Kepegawaian yaitu :

- a. Pegawai negeri adalah unsur aparatur negara, abdi negara, dan abdi masyarakat yang dengan kesetiaan dan ketaatan kepada Pancasila dan Undang-Undang Dasar 1945, negara dan pemerintah, menyelenggarakan tugas pemerintahan serta pembangunan.
- b. Pegawai negeri adalah mereka yang sudah memenuhi syarat-syarat yang ditentukan dalam peraturan perundang-undangan yang berlaku, diangkat oleh pejabat yang berwenang dan diberikan tugas negara lainnya yang ditetapkan

berdasarkan suatu peraturan perundang-undangan dan digaji menurut peraturan perundang-undangan yang berlaku.

#### **2.4 Metode Pengembangan Sistem**

Metode adalah suatu cara atau teknik sistematis untuk mengerjakan sesuatu. Pengembangan sistem adalah metode yang digunakan untuk mengembangkan suatu sistem informasi yang harus dikerjakan selama pengembangan sistem (Mall, 2003). Metode yang digunakan dalam pengembangan sistem pada sistem ini adalah metode *Waterfall*. Alasan menggunakan metode ini adalah karena metode ini melakukan pendekatan sistematis dan sekuensial dalam pengembangan perangkat lunak. Tahapan dari tiap metode ini harus diselesaikan terlebih dahulu sebelum masuk ketahapan berikutnya. Metode *Waterfall* merupakan metode yang sering digunakan oleh analis sistem pada umumnya. Metode *Waterfall* ini merupakan pengerjaan dari suatu sistem yang dilakukan secara berurutan atau secara linear. Jadi jika langkah *software requirements analysis* belum dikerjakan maka tidak bisa melakukan pengerjaan langkah *design, coding, testing* dan *maintenance* dengan menggunakan tahapan metode *Waterfall* (Mall, 2003).

#### **2.5 Pengertian *Unified Modelling Language* (UML)**

Sistem Informasi adalah kumpulan dari subsistem baik untuk fisik maupun nonfisik yang saling berhubungan satu sama lain dan bekerja sama untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berarti dan berguna (Susanto, 2004).

UML merupakan singkatan dari *Unified Modelling Language* adalah sekumpulan model konvensi yang digunakan untuk menentukan atau menggambarkan sebuah

sistem perangkat lunak (*software*) dalam kaitannya dengan objek (Whitten, Jeffery, L 2004:430). UML juga dapat diartikan sebuah bahasa grafik standar yang digunakan untuk memodelkan perangkat lunak berbasis objek.

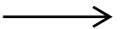
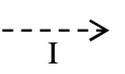
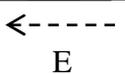
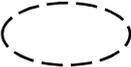
UML pertama kali dikembangkan pada pertengahan tahun 1990an dengan kerjasama antara James Rumbaugh, Grady Booch, dan Ivar Jacobson, yang masing-masing telah mengembangkan notasi UML diawal tahun 1990 (Timothy Lethbridge dan Robert Laganier, 2002:151).

## **2.6 Komponen-Komponen *Unified Modelling Language* (UML)**

### **2.6.1 *Use Case Diagram***

*Use Case Diagram* adalah salah satu jenis diagram pada UML dengan menggambarkan interaksi antara sistem dan aktor. *Use case* diagram dapat mendeskripsikan tipe antaraksi antara pemakai sistem dengan sistemnya. *Use Case* adalah deskripsi fungsi dari sebuah sistem yang dilihat dari sisi prespektif pengguna (Munawar, 2005). Simbol-simbol yang digunakan dalam merancang *use case* diagram ditunjukkan pada Tabel 2.1.

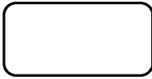
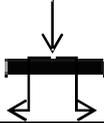
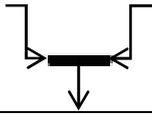
**Tabel 2.1** Simbol *Use Case Diagram*

Gambar	Nama	Keterangan
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
 Nama aktor	<i>Actor</i>	Menjelaskan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	<i>Association</i>	Menghubungkan antara objek satu dengan objek lainnya.
	<i>Directed Association</i>	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, <i>directed association</i> biasanya juga disertai dengan <i>multiplicity</i> .
	<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ) (umum-khusus).
	<i>Dependency</i>	Hubungan pada suatu elemen mandiri ( <i>independent</i> ) yang dapat mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
	<i>Include</i>	Menjelaskan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
	<i>Extend</i>	Menjelaskan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>System</i>	Menjelaskan paket yang menampilkan sistem secara terbatas.
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya ( <i>sinergi</i> ).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

### 2.6.2 Activity Diagram

*Activity Diagram* atau diagram aktivitas yaitu salah satu jenis diagram pada UML yang dapat memodelkan proses-proses yang terjadi pada sistem. Simbol-simbol yang dapat digunakan dalam merancang *Activity diagram* ditunjukkan pada Tabel 2.2.

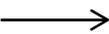
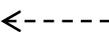
**Tabel 2.2** Simbol *Activity Diagram*

Gambar	Nama	Keterangan
	<i>Activity</i>	Menandakan sebuah aktivitas. Bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi.
	<i>Initial Node</i>	Titik awal, untuk memulai suatu aktivitas.
	<i>Activity Final Node</i>	Objek dibentuk dan diselesaikan atau diakhiri.
	<i>Fork</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.
	<i>Join</i>	Beberapa aliran yang pada tahap tertentu berubah menjadi satu aliran.
	<i>Decision</i>	Pilihan untuk mengambil keputusan.

### 2.6.3 Sequence Diagram

*Sequence* Diagram yaitu salah satu jenis pada UML yang menjelaskan interaksi objek yang berdasarkan urutan waktu, *sequence* diagram juga dapat menggambarkan urutan atau tahapan yang harus dilakukan untuk dapat menghasilkan suatu *use case* diagram. Simbol-simbol yang digunakan dalam merancang *Sequence* diagram ditunjukkan pada Tabel 2.3.

**Tabel 2.3** Simbol *Sequence* Diagram

Gambar	Nama	Keterangan
	<i>Lifeline</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi - informasi tentang aktifitas yang terjadi.
	<i>Self Message</i>	<i>Message</i> yang diberikan oleh sebuah objek kepada dirinya sendiri.
	<i>Async Message</i>	Menghubungkan dua objek, pesan asinkron tidak perlu balasan agar interaksi berlanjut.
	<i>Replay Message</i>	<i>Message</i> yang diberikan oleh sebuah objek untuk membalas <i>message</i> objek lain.
	<i>Create Message</i>	<i>Message</i> yang diberikan oleh sebuah objek untuk membuat objek baru.
	<i>Delete Message</i>	<i>Message</i> yang diberikan oleh sebuah objek untuk menghapus objek.

### 2.6.4 Class Diagram

*Class Diagram* yaitu salah satu jenis pada UML yang digunakan untuk menampilkan kelas-kelas yang ada pada suatu sistem yang digunakan. Jadi diagram ini dapat memberikan sebuah gambaran mengenai sistem maupun relasi- relasi yang terdapat pada sistem tersebut. Simbol-simbol yang digunakan dalam merancang *Class diagram* ditunjukkan pada Tabel 2.4.

**Tabel 2.4** Simbol *Class Diagram*

Gambar	Nama	Keterangan
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Association</i>	Menghubungkan antara objek satu dengan objek lainnya.
	<i>Directed Association</i>	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, <i>directed association</i> biasanya juga disertai dengan <i>multiplicity</i> .
	<i>Aggregation</i>	Relasi antar kelas dengan makna semua- bagian ( <i>Whole-part</i> ).
	<i>Composition</i>	Jika sebuah class tidak bisa berdiri sendiri dan harus merupakan bagian dari class yang lain, maka class tersebut memiliki relasi <i>composition</i> terhadap class tempat dia bergantung tersebut.
	<i>Dependency</i>	Hubungan pada suatu elemen mandiri ( <i>independent</i> ) yang dapat mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
	<i>Generalization</i>	Hubungan objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ) (umum-khusus).

## 2.7 Pengertian Website

*Website* merupakan kumpulan halaman *Web* yang saling terhubung dan file- filenya saling terkait. *Web* terdiri dari *page* atau halaman, dan kumpulan halaman yang dinamakan *homepage*. *Homepage* berada pada posisi teratas, dengan halaman-halaman terkait berada dibawahnya. Biasanya setiap halaman dibawah *homepage* disebut *child page*, yang berisi *hyperlink* ke halaman lain dalam *Web* (Gregorius, 2000:30). *Website* awalnya merupakan suatu layanan sajian informasi yang menggunakan konsep *hyperlink*, yang memudahkan *surfer* atau pengguna *internet* melakukan penelusuran informasi di *internet*. Informasi yang disajikan dengan *Web* menggunakan konsep *multimedia*, informasi dapat disajikan dengan menggunakan banyak media, seperti teks, gambar, animasi, suara, atau film.

## 2.8 Software Pendukung Pengembangan Sistem Kepegawaian

*Software* yang digunakan dalam pembangunan sistem informasi kepegawaian ini adalah PHP (*Hypertext Preprocessor*) *script* yang memberikan algoritma sebelum ditampilkan ke halaman *Web*, *MySQL* sebagai *software database*, XAMPP sebagai kumpulan dari berbagai *software* dan *dreamweaver* sebagai *software* pengatur desain tampilan sistem.

### 2.8.1 PHP (*Hypertext Preprocessor*)

PHP merupakan bahasa *scripting* (bahasa pemrograman) yang menyatu dengan HTML dan berada pada *server* (*server side HTML embedded scripting*). Dengan menggunakan PHP dapat membuat beragam aplikasi berbasis *web*, mulai dari halaman *web* yang sederhana sampai aplikasi kompleks yang membutuhkan koneksi ke *database* (Tulangow, 2011). PHP adalah bahasa *scripting* yang

menyatu dengan HTML dan dijalankan pada *server side* yang artinya semua *sintaks* yang diberikan sepenuhnya dijalankan pada *server* sedangkan yang dikirimkan ke *browser* hanya hasilnya saja.

Bahasa pemrograman PHP ini membuat orang lain tidak dapat mengetahui kode sumber yang digunakan untuk mendesain *web*, karena yang ditampilkan di *browser* pada sisi *client* adalah *file* hasil eksekusi dari *server*, sedangkan *file* program phpnya hanya dapat dilihat di sisi *server* (Ramadhan dan Nugroho, 2009).

### 2.8.2 Laravel

Laravel merupakan *web application framework* berbasis PHP yang bersifat *open source*, menggunakan konsep *model-view-controller* (MVC). Laravel berada di bawah lisensi MIT *License*, dengan menggunakan *GitHub* sebagai tempat berbagi kode (Awaludin, 2016).

Laravel merupakan *framework* PHP yang menekankan pada kesederhanaan serta fleksibilitas pada desainnya. Laravel dirilis di bawah lisensi MIT dengan sumber kode yang disediakan di *Github*. Sama seperti *framework* PHP lainnya, Laravel dibangun dengan basis MVC (*Model- View-Controller*). Laravel dilengkapi *command line tool* yang bernama “Artisan” yang bisa digunakan untuk *packaging bundle* dan instalasi *bundle* (Rohman, 2014).

*Framework* Laravel dibuat oleh Taylor Otwell, proyek laravel dimulai pada April 2011. Laravel dibuat karena Otwell tidak dapat menemukan *framework* yang *up-to-date*. Laravel adalah *framework* yang menganut pola *Model- View-Controller*

(MVC) dan merupakan salah satu *framework PHP* yang populer saat ini. *Laravel* juga dapat membuat paket pada *framework laravel* serta dapat mengemas paket tampilan pada *framework laravel*. *Laravel* memerlukan beberapa kebutuhan untuk menjalankan *frameworknya* yaitu dengan memilih *text editor* sesuai selera pengguna, kemudian *web server* yang mendukung versi *laravel* terbaru, dan *composer* berfungsi sebagai penginstal *third-party* untuk aplikasi *web* secara cepat (Triyono, 2016).

Beberapa fitur yang dimiliki oleh *framework laravel* adalah sebagai berikut:

- a. *Bundles* yaitu sebuah fitur dengan sistem pengemasan modular dan berbagai *bundle* telah tersedia untuk digunakan dalam aplikasi.
- b. *Eloquent ORM* merupakan penerapan PHP lanjutan dari pola “*active record*” menyediakan metode internal untuk mengatasi kendala hubungan antara objek *database*. Pembangun *query laravel fluent* didukung *Eloquent*.
- c. *Application Logic* merupakan bagian dari aplikasi yang dikembangkan, baik menggunakan *controllers* maupun sebagai bagian dari deklarasi *route*. *Sintaks* yang digunakan untuk mendefinisikannya sama dengan yang digunakan oleh *framework sinatra*.
- d. *Restful Controllers*, memberikan sebuah *option* (pilihan) untuk memisahkan logika dalam melayani HTTP *GET* dan permintaan *POST*.
- e. *Class Auto Loading*, menyediakan otomatis *loading* untuk *class-class* PHP, tanpa membutuhkan pemeriksaan manual terhadap jalur masuknya. Fitur ini mencegah *loading* yang tidak perlu.
- f. *View Composers* adalah kode unit *logical* yang dapat dijalankan ketika sebuah *View* di-load.

- g. *IoC Container* memungkinkan untuk objek baru yang dihasilkan dengan mengikuti prinsip *control* pembalik, dengan pilhan contoh dan referensi dari objek baru sebagai *Singletons*.
- h. *Migrations* menyediakan versi sistem *control* untuk skema *database*, sehingga memungkinkan untuk menghubungkan perubahan adalah basis kode aplikasi dan keperluan yang dibutuhkan dalam merubah tata letak *database*, mempermudah dalam penempatan dan memperbarui aplikasi.
- i. *Unit Testing* mempunyai peran penting dalam *framework* laravel, *unit testing* ini mempunyai banyak tes untuk medeteksi dan mencegah regresi. *Unit testing* dapat dijalankan melalui fitur “*artisan command -line*”.
- j. *Automatic pagination* menyederhanakan tugas dari penerapan halaman, menggantikan penerapan yang manual dengan metode otomatis yang ter-integrasi ke laravel.
- k. *Reverse Routing*, mendefinisikan hubungan antara *Link* dan *Route*, sehingga jika suatu saat ada perubahan pada *route* secara otomatis akan tersambung dengan *link* yang relevan. Ketika *link* yang dibuat dengan menggunakan nama- nama dari *route* yang ada, secara otomatis laravel akan membuat URL yang sesuai (Aminudin, 2015).

Kelebihan *framework* laravel adalah sebagai berikut:

a. *Expressif*

Laravel adalah *framework PHP* yang *expressif*, artinya ketika melihat suatu *sintaks programmer* akan tahu kegunaannya meskipun belum pernah mempelajari atau menggunakan laravel.

### *b. Simple*

Laravel memiliki *Eloquent ORM (Object Relation Mapping)* yaitu fitur untuk memanggil data secara mudah.

### *c. Accessible*

Laravel dibuat dengan dokumentasi yang lengkap sehingga ketika laravel merilis versi barunya laravel berkomitmen untuk selalu menyertakan dokumentasi yang lengkap (Aminudin, 2015).

## **2.8.3 Definisi MVC (*Model View Controller*)**

*Model-View-Controller* (MVC) pertama kali diperkenalkan peneliti *Xerox PARC* pada bidang penelitian bahasa pemrograman *Smalltalk* akhir tahun 1970-an.

*Model-View-Controller* adalah sebuah pola perancangan perangkat lunak yang dibangun dengan tiga jenis komponen utama yaitu (Pitt, 2012):

- a. Model berfungsi untuk membantu pengelolaan data.
- b. View berfungsi untuk mengatur tampilan ke pengguna.
- c. Controller berisi perintah-perintah untuk memproses suatu data dan mengirimkannya ke halaman web.

## **2.8.4 Framework**

Secara umum, *Framework* menggunakan struktur MVC (*Model, View, Controller*).

*Framework* adalah sekumpulan *library* yang diorganisasikan pada sebuah rancangan arsitektur untuk memberikan kecepatan, ketepatan, kemudahan dan konsistensi di dalam pengembangan aplikasi dari definisi tersebut” (Triyono, 2016).

*Framework* terdiri dari:

*a. Model*

Model mencakup semua proses yang terkait dengan pemanggilan struktur data baik berupa pemanggilan fungsi, *input processing* atau mencetak *output* ke dalam *browser*.

*b. View*

*View* mencakup semua proses yang terkait *layout output*. Bisa dibilang untuk meletakkan *template interface website* atau aplikasi.

*c. Controller*

*Controller* mencakup semua proses yang terkait dengan pemanggilan *database* dan kapsulisasi proses-proses utama. Jadi dibagian ini ada *file* bernama *member.php*, maka semua proses yang terkait dengan *member* akan dikapsulisasi atau dikelompokkan dalam *file* ini.

Kelebihan dengan adanya *Framework* akan lebih mempermudah memahami mekanisme kerja dari sebuah aplikasi. Ini tentunya akan sangat membantu proses pengembangan sistem yang dilakukan secara *team*.

### **2.8.5 MySQL**

Pada awal perkembangannya disebut *SQL* yang merupakan kepanjangan dari *Structured Query Language*. *SQL* merupakan bahasa standar yang digunakan untuk mengakses *database server* dalam konteks bahasa *SQL*. Pada umumnya informasi tersimpan dalam tabel-tabel yang secara logika merupakan struktur dua dimensi yang terdiri atas baris-baris data yang berada dalam kolom-kolom atau lebih kolom. Baris pada tabel disebut *instance* dalam data, sedangkan kolom sering disebut sebagai *attribute* atau *field*.

Keseluruhan tabel itu dihimpun dalam satu kesatuan yang disebut *database*. *MySQL* adalah salah satu program yang dapat digunakan sebagai *database*, dan merupakan salah satu *software* untuk *database server* yang banyak digunakan. *MySQL* bersifat *open source* dan menggunakan SQL. *MySQL* bisa dijalankan di berbagai *platform* misalnya *Windows*, *Linux*. *MySQL* adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (*General Public License*).

Pengguna dapat secara bebas menggunakan *MySQL*, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. *MySQL* merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya. SQL (*Structured Query Language*) adalah sebuah konsep pengoperasian basis data terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis (Februariyanti dan Zuliarso, 2012).

### **2.8.6 XAMPP**

XAMPP adalah sebuah *software Web server apache* yang didalamnya sudah tersedia *database server MySQL* dan dapat mendukung pemrograman PHP. XAMPP merupakan *software* yang mudah digunakan, gratis serta mendukung instalasi di *Linux* dan *Windows*. Keuntungan lainnya adalah cuma menginstal satu kali sudah tersedia *Apache Web Server*, *MySQL Database Server*, *PHP Support* (PHP 4 dan PHP 5) dan beberapa *module* lainnya (Februariyanti dan Zuliarso, 2012).

### **2.8.7 Dreamweaver**

*Dreamweaver* adalah sebuah HTML untuk mendesain *Web* secara *visual* dan mengelola situs atau halaman *Web*. *Dreamweaver* merupakan *software* utama yang digunakan oleh *Web Designer* maupun *Web Programmer* dalam mengembangkan suatu situs *Web*, karena *dreamweaver* mempunyai ruang kerja, fasilitas dan kemampuan yang mampu meningkatkan produktivitas dan efektivitas dalam *desain* maupun membangun suatu situs *Web* (Madcoms, 2011).

### **2.8.8 Database**

*Database* adalah kumpulan fakta-fakta sebagai representasi dari dunia nyata yang saling berhubungan dan mempunyai arti tertentu. *Database* dapat menangani data mulai dari data sederhana sampai data yang rumit. Data dalam skala kecil sampai dengan data yang dalam skala besar, untuk satu pengguna atau banyak pengguna, yang disebut dengan kompleksitas *database* (Robi'in, 2002). Dari beberapa pendapat di atas maka dapat dikatakan bahwa *database* adalah kumpulan fakta-fakta data yang saling berhubungan satu dengan lainnya dan menyimpan suatu koleksi data. Menurut Simarmata (2007) *Database* adalah suatu aplikasi terpisah yang menyimpan suatu koleksi data. Sebuah *database* adalah tempat penyimpanan *file* data, sebagai *file* data, dan *database* tidak menyajikan informasi secara langsung kepada pengguna.

### **2.8.9 HTML (Hypertext Markup Language)**

HTML (*Hypertext Markup Language*) merupakan bahasa yang digunakan dalam pembuatan *web* dengan *hypertext* dan informasi lain yang akan ditampilkan pada halaman *web*. *File* HTML dapat berisi teks, gambar, dan tipe informasi lainnya

seperti data *file*, *audio* dan program *executeable*. *File* HTML berisi instruksi yang diterjemahkan ke *browser* yang ada di komputer pengguna. Pengguna dapat mengontrol tampilan dan kontennya sesuai instruksi *file* di HTML, mempublikasikan *file* secara *online* yang dapat diakses dunia (Sutarman, 2012).

#### **2.8.10 Definisi *Black Box Testing***

Ada dua macam pendekatan kasus uji yaitu *white-box* dan *black-box*. Pendekatan *white-box* adalah pengujian untuk memperlihatkan cara kerja dari produk secara rinci sesuai dengan spesifikasinya akan dites dengan menyediakan kasus uji yang akan mengerjakan kumpulan kondisi dan pengulangan secara spesifik. Sehingga melalui penggunaan metode ini akan dapat memperoleh kasus uji yang menjamin bahwa semua jalur independen pada suatu model telah digunakan minimal satu kali, penggunaan keputusan logis pada sisi benar dan salah, pengeksekusian semua *loop* dalam batasan dan batas operasional perekayasa, serta penggunaan struktur data internal guna menjamin validitasnya (Pressman, 2010).

Pendekatan *black-box* merupakan pendekatan pengujian untuk mengetahui apakah semua fungsi perangkat lunak telah berjalan semestinya sesuai dengan kebutuhan fungsional yang telah didefinisikan (Jiang & Lu, 2012). Kasus ini bertujuan untuk menunjukkan fungsi perangkat lunak tentang cara beroperasinya. Teknik pengujian ini berfokus pada domain informasi dari perangkat lunak, yaitu melakukan kasus uji dengan mempartisi *domain input* dan *output* program.

Metode *black-box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi *input* yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian ini berusaha menemukan kesalahan

dalam kategori fungsi-fungsi yang tidak benar atau hilang, kesalahan *interface*, kesalahan dalam struktur data atau akses basis data eksternal, kesalahan kinerja, dan inisialisasi dan kesalahan terminal (Pressman, 2010).

### **III. METODE PENELITIAN**

#### **3.1 Waktu dan Tempat Penelitian**

Penelitian ini dilaksanakan di Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung yang berada di Jl. Soemantri Brojonegoro No.1 Gedung Meneng, Bandar Lampung. Waktu pelaksanaan penelitian ini dimulai pada bulan Januari 2019 sampai dengan selesai.

#### **3.2 Alat Pendukung**

Penelitian ini dilakukan dengan menggunakan alat yang mendukung dan menunjang pelaksanaan penelitian, yaitu:

##### 1. Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan dalam pengembangan system informasi pemberkasan perkuliahan ini adalah 1 unit laptop dengan spesifikasi:

- a) *Processor* : Intel(R) Core(TM) i3-4010U CPU@1.70GHz (4CPUs), ~1.7GHz
- b) *Installed memory (RAM)* : 4096 MB
- c) *System type* : 64-bit *Operating System*

##### 2. Perangkat Lunak

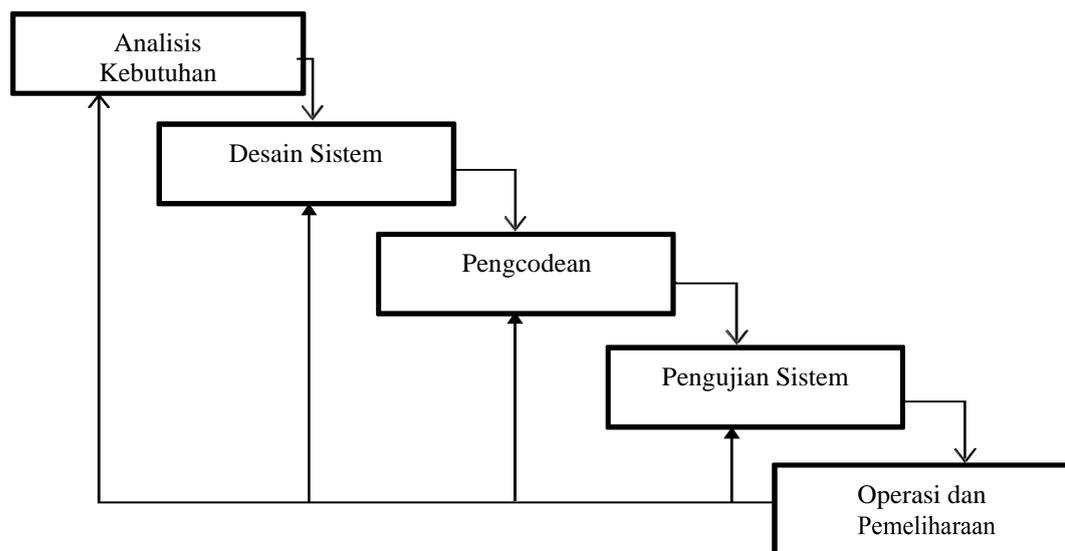
(*Software*) yang digunakan dalam pengembangan sistem ini adalah:

- a) *Operating System* : Windows 10 Pro 64-Bit
- b) *Text Editor Visual Studio Code*
- c) *Mariadb* sebagai basis data

- d) *Apache* versi 2.4.29 sebagai *web server*
- e) *Corel Draw X7* untuk mendesain rancangan *interface system*
- f) *Web Browser Mozilla Firefox* dan *Google Chrome*
- g) *Star UML* untuk mendesain rancangan sistem
- h) *Laravel Framework*

### 3.3 Tahapan Penelitian

Metode yang digunakan dalam perancangan sistem informasi kepegawaian ini adalah menggunakan metode *Waterfall*. Dasar penggunaan metode ini adalah karena metode *waterfall* melakukan pendekatan secara sistematis dan berurutan dalam membangun suatu sistem. Proses metode *Waterfall* yaitu pada pengerjaan dari suatu sistem dilakukan secara berurutan. Sesuai dengan rancangan sistem yang dihasilkan berkualitas baik, dikarenakan pelaksanaannya secara bertahap sehingga tidak terfokus pada tahapan tertentu. Tahapan dari metode *Waterfall* dapat dilihat pada Gambar 3.1.



**Gambar 3.1** Metode *Waterfall*

Pada Gambar 3.1 Metode *Waterfall* memiliki beberapa tahapan yang berurut yakni, *Requirement* (Analisis Kebutuhan), *Design System* (Desain Sistem), *Coding* (Pengkodean) & *Testing* (Pengujian), Penerapan Program dan Pemeliharaan (*Operation & Maintenance*). Tahapan-tahapan dari metode *waterfall* adalah sebagai berikut :

#### 1. *Requirement Analisis* (Analisis Kebutuhan)

Pada tahap ini analisis kebutuhan untuk rancang sistem ini diperlukan perangkat lunak (*software*) dan perangkat keras (*hardware*) yang diharapkan oleh pengguna. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna. Analisis kebutuhan perangkat keras pada sistem ini yaitu laptop dengan spesifikasi sebagai berikut Acer Aspire ES1-411, HDD 500GB, *Processor Intel Celeron*, RAM 2 GB. Analisis kebutuhan perangkat lunak yang membantu pembuatan sistem ini yaitu *Operating System Windows 10 Pro*, *Visual Studio Code*, StarUML, XAMPP versi 1.7.3, Bahasa Pemrograman PHP, *Mozilla firefox 9.0*, dan *MySQL Server 64 Bit*.

#### 2. *System Design* (Desain Sistem)

Spesifikasi kebutuhan dari tahap sebelumnya dipelajari dalam fase ini dan desain sistem disiapkan. Desain Sistem membantu dalam mendefinisikan arsitektur sistem secara keseluruhan. Tahap ini dibuat sebelum tahap pengkodean. Dokumentasi yang dihasilkan dari tahap desain sistem ini antara lain perancangan *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram* dan Perancangan *Interface*.

### 3. *Coding* (Pengkodean Program)

Aktivitas pada tahap ini dilakukan pengkodean sistem. Penulisan kode program merupakan tahap penerjemahan desain sistem yang telah dibuat ke dalam bentuk perintah-perintah yang dimengerti komputer dengan mempergunakan bahasa pemrograman. Tahapan ini merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Sistem ini bahasa pemrograman yang dipakai adalah *PHP* dan *Database MySQL*.

### 4. *Testing* (Pengujian Program)

Seluruh *unit* yang dikembangkan dalam tahap implementasi diintegrasikan ke dalam sistem setelah pengujian yang dilakukan masing-masing *unit*. Setelah integrasi seluruh sistem, diuji untuk mengecek setiap kegagalan maupun kesalahan.

### 5. *Operation & Maintenance* (Operasi dan Pemeliharaan)

Tahap akhir yang terdapat dalam metode *waterfall*. Perangkat lunak yang sudah jadi, dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perbaikan implementasi *unit* sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

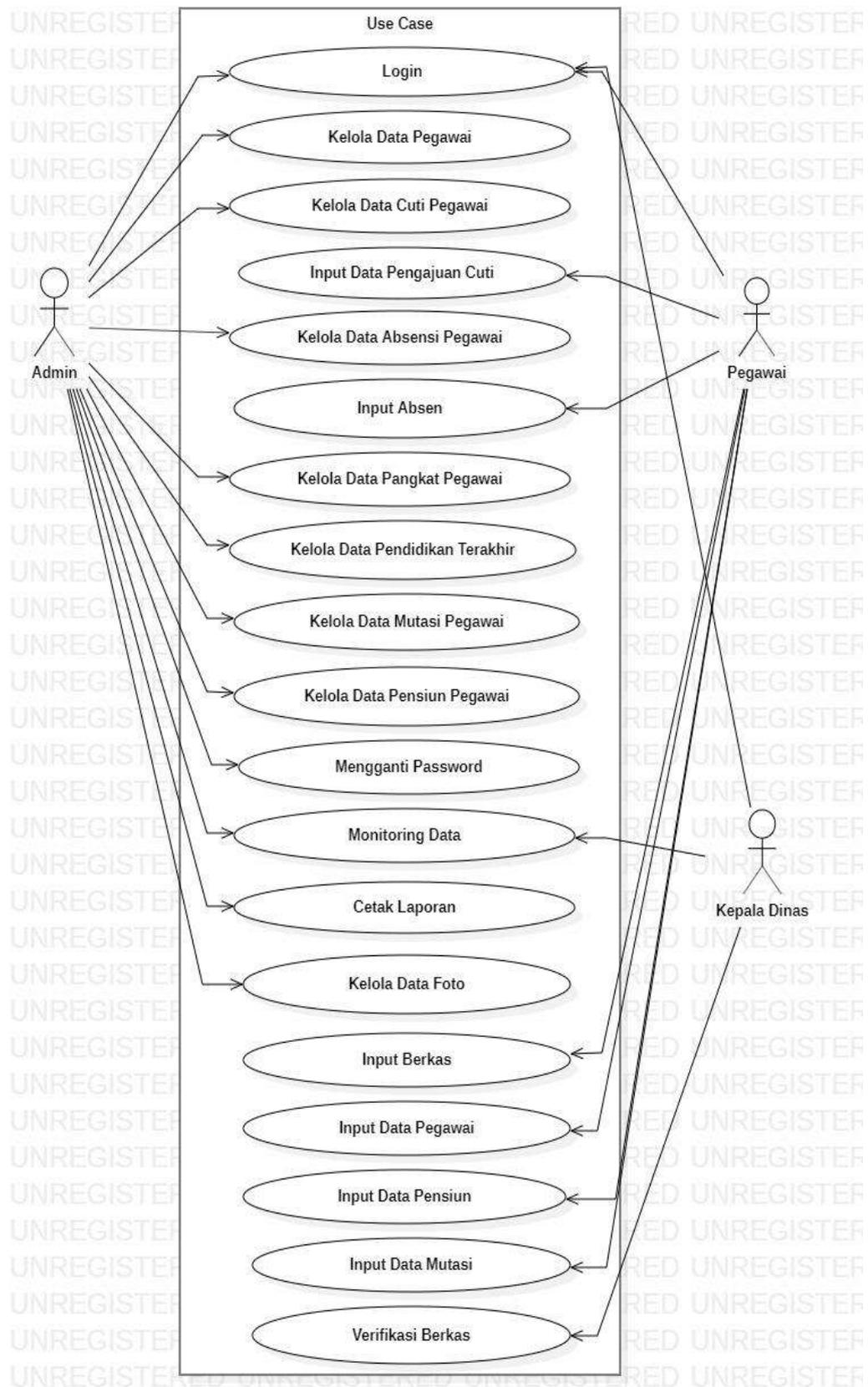
### 3.4 Pengajuan Solusi Alternatif

Untuk menemukan solusi alternatif agar dapat mengatasi masalah yang ada, maka pada penelitian ini menggunakan beberapa metode yang digunakan sehingga masalah teratasi. Pada perancangan ini digunakan UML ( *Unified Modelling Language* ) untuk membuat suatu sistem informasi yang menjelaskan proses sistem informasi itu berjalan.

#### 3.4.1 Use Case Diagram

Untuk menganalisis sistem informasi digunakan *use case* diagram agar mempermudah untuk membangun sebuah sistem. Aktor menggambarkan segala pengguna *software* aplikasi. Aktor dalam Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta. *Admin* dapat mengakses sistem untuk melakukan pekerjaannya dalam pengelolaan keseluruhan data pegawai. *Admin* memiliki hak akses penuh untuk mengelola sistem ini dan berhak mengatur dan menjalankannya.

*Use Case Diagram* yang diterapkan pada Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta dapat dilihat pada Gambar 3.2.



**Gambar 3.2** Use Case Diagram Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta

### 3.4.1.1 Definisi Aktor

Definisi aktor merupakan penjelasan yang dilakukan oleh aktor-aktor yang terlibat dalam sistem informasi ini. Skripsi dari aktor-aktor yang terlibat perancangan Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta disajikan pada Tabel 3.1.

**Tabel 3.1** Definisi Aktor

No.	Aktor	Deskripsi
1.	<i>Admin</i>	<ol style="list-style-type: none"> <li>1. Mengelola program-program yang telah diusulkan didalam sistem.</li> <li>2. Mengelola kegiatan-kegiatan dari program yang telah diusulkan.</li> </ol>
2.	Pegawai	<ol style="list-style-type: none"> <li>1. Melakukan <i>input</i> data pegawai.</li> <li>2. Mengisi data pada form yang sudah ada pada sistem.</li> </ol>
3.	Kepala Dinas	<ol style="list-style-type: none"> <li>1. Melakukan Verifikasi Berkas.</li> </ol>

### 3.4.1.2 Definisi Use Case

Definisi *use case* merupakan penjelasan dari kegiatan dalam sistem informasi ini. Skripsi dari *use case* dalam perancangan Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta disajikan pada Tabel 3.2.

**Tabel 3.2** Definisi Use Case Diagram

No.	Nama Use Case	Defenisi	Aktor
1.	Kelola Data Pegawai	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam mengelola data pegawai yang telah diusulkan	<i>Admin</i>
2.	Kelola Data Cuti Pegawai	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam mengelola data cuti pegawai yang telah diusulkan	<i>Admin</i>
3.	Kelola Data Absensi Pegawai	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam mengelola data absensi pegawai yang telah diusulkan	<i>Admin</i>
4.	Kelola Data Pangkat Pegawai	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam mengelola data pangkat pegawai yang telah diusulkan	<i>Admin</i>

**Tabel 3.2** Definisi *Use Case Diagram* (Lanjutan)

No.	Nama <i>Use Case</i>	Defenisi	Aktor
5.	Kelola Data Pendidikan Terakhir	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam mengelola data pendidikan terakhir yang telah diusulkan	<i>Admin</i>
6.	Kelola Data Mutasi Pegawai	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam mengelola data mutasi pegawai yang telah diusulkan	<i>Admin</i>
7.	Kelola Data Pensiun Pegawai	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam mengelola data pensiun pegawai yang telah diusulkan	<i>Admin</i>
8.	Kelola Data Foto	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam mengelola dan merekap data foto	<i>Admin</i>
9.	Cetak Laporan	<i>Use Case</i> ini difungsikan untuk mencetak laporan yang telah dipilih	<i>Admin</i>
10.	<i>Monitoring Data</i>	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam menyeleksi dan memperhatikan data	<i>Admin - Kadis</i>
11.	Mengganti <i>Password</i>	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> untuk mengganti <i>password</i> serta memverifikasi	<i>Admin</i>
12.	<i>Input Data Pengajuan Cuti</i>	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam melakukan <i>input data pengajuan cuti</i>	Pegawai
13.	<i>Input Absen</i>	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam melakukan <i>input absen</i>	Pegawai
14.	<i>Input Foto</i>	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam melakukan <i>input file foto</i>	Pegawai
17.	<i>Input Data Pegawai</i>	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam melakukan <i>input data pegawai</i>	Pegawai
18.	<i>Input Data Pensiun</i>	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam melakukan <i>input data pensiun</i>	Pegawai
19.	<i>Input Data Mutasi</i>	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam melakukan <i>input data mutasi</i>	Pegawai
20.	<i>Verifikasi Berkas</i>	<i>Use Case</i> ini menggambarkan kegiatan <i>user</i> dalam melakukan <i>Verifikasi berkas</i>	Kepala Dinas

### 3.4.1.3 Skenario *Use Case Diagram*

Pada skenario *use case* dilakukan secara berurutan kegiatan yang dilakukan oleh sistem dan aktor, yang disajikan pada Tabel 3.3 sampai dengan 3.25.

**Tabel 3.3** Skenario *Use Case Login* sebagai Pegawai

Aksi Aktor	Respon Sistem
1. Pegawai memasukkan <i>username</i> dan <i>password</i>	
	2. Sistem mengecek <i>username</i> dan <i>password</i> , jika <i>username</i> dan <i>password</i> salah
3. Pegawai memasukkan <i>username</i> dan <i>password</i> Kembali	
	4. Sistem mengecek <i>username</i> dan <i>password</i> , jika <i>username</i> dan <i>password</i> benar
	5. Sistem menampilkan <i>menu</i> utama pegawai

**Tabel 3.4** Skenario *Use Case Kelola Data* Pegawai

Aksi Aktor	Respon Sistem
1. <i>Admin</i> mengakses <i>menu</i> kelola data pegawai	
	2. Sistem menampilkan halaman kelola data pegawai
3. <i>Admin</i> mengelola data pegawai	
	4. Sistem menambah data Pegawai
	5. Sistem mengedit data pegawai
	6. Sistem menghapus data Pegawai
	7. Sistem mencari data pegawai

**Tabel 3.5** Skenario *Use Case* Kelola Data Cuti Pegawai

Aksi Aktor	Respon Sistem
1. <i>Admin</i> mengakses <i>menu</i> data cuti Pegawai	
	2. Sistem menampilkan halaman kelola data cuti pegawai
3. <i>Admin</i> mengelola data cuti pegawai	
	4. Sistem mengedit data cuti pegawai
	5. Sistem menghapus data cuti pegawai
	6. Sistem mencari data cuti pegawai
	7. Sistem menambah data cuti pegawai

**Tabel 3.6** Skenario *Use Case* Input Data Pengajuan Cuti

Aksi Aktor	Respon Sistem
1. Pegawai memilih <i>menu input</i> data pengajuan cuti	
	2. Sistem menampilkan halaman <i>input</i> data pengajuan cuti
3. Pegawai mengisi data pengajuan cuti	
	4. Sistem menambah data pengajuan cuti

**Tabel 3.7** Skenario *Use Case* Login sebagai *Admin*

Aksi Aktor	Respon Sistem
1. <i>Admin</i> memasukkan <i>username</i> dan <i>password</i>	
	2. Sistem mengecek <i>username</i> dan <i>password</i> , jika <i>username</i> dan <i>password</i> salah
3. <i>Admin</i> memasukkan <i>username</i> dan <i>password</i> Kembali	
	4. Sistem mengecek <i>username</i> dan <i>password</i> , jika <i>username</i> dan <i>password</i> benar
	5. Sistem menampilkan <i>menu</i> utama <i>admin</i>

**Tabel 3.8** Skenario *Use Case* Kelola Data Absensi Pegawai

Aksi Aktor	Respon Sistem
1. <i>Admin</i> mengakses <i>menu</i> data absensi pegawai	
	2. Sistem menampilkan halaman data absensi pegawai
3. <i>Admin</i> mengelola data absensi pegawai	
	4. Sistem menambah data absensi pegawai
	5. Sistem menghapus data absensi pegawai
	6. Sistem <i>mengedit</i> data absensi Pegawai
	7. Sistem mencari data absensi Pegawai

**Tabel 3.9** Skenario *Use Case* Input Absen

Aksi Aktor	Respon Sistem
1. Pegawai memilih <i>menu input</i> absen	
	2. Sistem menampilkan halaman <i>input</i> absen
3. Pegawai mengisi absen	
	4. Sistem menambah absen

**Tabel 3.10** Skenario *Use Case* Kelola Data Pangkat Pegawai

Aksi Aktor	Respon Sistem
1. <i>Admin</i> mengakses <i>menu</i> data pangkat pegawai	
	2. Sistem menampilkan halaman data pangkat pegawai
3. <i>Admin</i> mengelola data pangkat pegawai	
	4. Sistem menambah data pangkat pegawai
	5. Sistem <i>mengedit</i> data pangkat pegawai
	6. Sistem menghapus data pangkat pegawai
	7. Sistem mencari data pangkat pegawai

**Tabel 3.11** Skenario *Use Case* Kelola Data Pendidikan Terakhir

Aksi Aktor	Respon Sistem
1. <i>Admin</i> mengakses <i>menu</i> data pendidikan terakhir	
	2. Sistem menampilkan halaman data pendidikan terakhir
3. <i>Admin</i> mengelola data pendidikan terakhir	
	4. Sistem menambah data pendidikan terakhir
	5. Sistem <i>mengedit</i> pendidikan terakhir
	6. Sistem menghapus data pendidikan terakhir
	7. Sistem mencari data pendidikan terakhir

**Tabel 3.12** Skenario *Use Case* Kelola Data Mutasi Pegawai

Aksi Aktor	Respon Sistem
1. <i>Admin</i> mengakses data mutasi Pegawai	
	2. Sistem menampilkan halaman data mutasi pegawai
3. <i>Admin</i> mengelola data mutasi Pegawai	
	4. Sistem menambah data mutasi pegawai
	5. Sistem <i>mengedit</i> data mutasi pegawai
	6. Sistem menghapus data mutasi pegawai
	7. Sistem mencari data mutasi pegawai

**Tabel 3.13** Skenario *Use Case* Kelola Data Pensiun Pegawai

Aksi Aktor	Respon Sistem
1. <i>Admin</i> mengakses data pensiun pegawai	
	2. Sistem menampilkan halaman data pensiun pegawai
3. <i>Admin</i> mengelola data pensiun pegawai	
	4. Sistem menambah data pensiun pegawai
	5. Sistem mencari data pensiun pegawai
	6. Sistem menghapus data pensiun pegawai
	7. Sistem <i>mengedit</i> data pensiun pegawai

**Tabel 3.14** Skenario *Use Case* Input Data Pensiun

Aksi Aktor	Respon Sistem
1. Pegawai memilih <i>menu input</i> data pension	
	2. Sistem menampilkan halaman data pension
3. Pegawai mengisi data pensiun	
	4. Sistem menambah data pension

**Tabel 3.15** Skenario *Use Case* Cetak Laporan

Aksi Aktor	Respon Sistem
1. <i>Admin</i> mengakses cetak laporan	
	2. Sistem menampilkan halaman cetak laporan
3. <i>Admin</i> memilih <i>button print</i> atau <i>Eksport</i>	
	4. Sistem mencetak laporan

**Tabel 3.16** Skenario *Use Case Monitoring Data*

Aksi Aktor	Respon Sistem
1. <i>Admin</i> mengakses <i>menu monitoring data</i>	
	2. Sistem menampilkan halaman <i>monitoring data</i>
3. <i>Admin</i> memilih dan mencari data	
	4. Sistem menampilkan data yang dicari dan terpilih

**Tabel 3.17** Skenario *Use Case Kelola Data Foto*

Aksi Aktor	Respon Sistem
1. <i>Admin</i> mengakses <i>menu data foto</i>	
	2. Sistem menampilkan halaman data foto
3. <i>Admin</i> mengelola data foto	
	4. Sistem menambah data foto
	5. Sistem mencari data foto
	6. Sistem menghapus data foto

**Tabel 3.18** Skenario *Use Case Input Foto*

Aksi Aktor	Respon Sistem
1. Pegawai memilih <i>menu input foto</i>	
	2. Sistem menampilkan halaman <i>input foto</i>
3. Pegawai memasukkan foto	
	4. Sistem menambah foto

**Tabel 3.19** Skenario *Use Case Input Data Pegawai*

Aksi Aktor	Respon Sistem
1. Pegawai memilih <i>menu input data pegawai</i>	
	2. Sistem menampilkan halaman <i>input data pegawai</i>
3. Pegawai memasukkan data pegawai	
	4. Sistem menambah data pegawai

**Tabel 3.20** Skenario *Use Case* Mengganti *Password*

Aksi Aktor	Respon Sistem
1. <i>Admin</i> mengakses <i>menu</i> ganti <i>Password</i>	
	2. Sistem menampilkan halaman ganti <i>password</i>
3. <i>Admin</i> memasukkan <i>username</i> dan <i>password</i> lama	
4. <i>Admin</i> memasukkan <i>password</i> baru dan konfirmasi <i>password</i> baru	
	5. Sistem mengkonfirmasi <i>password</i> baru
6. <i>Admin</i> mengklik <i>button</i> simpan	
	7. Sistem menyimpan <i>password</i> baru

**Tabel 3.21** Skenario *Use Case* Kelola Data Pegawai

Aksi Aktor	Respon Sistem
1. <i>Admin</i> mengakses <i>menu</i> kelola data pegawai	
	2. Sistem menampilkan halaman kelola data pegawai
3. <i>Admin</i> mengelola data pegawai	
	4. Sistem menambah data pegawai
	5. Sistem mengedit data pegawai
	6. Sistem menghapus data pegawai
	7. Sistem mencari data pegawai

**Tabel 3.22** Skenario *Use Case* *Input* Data Mutasi Pegawai

Aksi Aktor	Respon Sistem
1. Pegawai memilih <i>menu</i> <i>input</i> data mutasi pegawai	
	2. Sistem menampilkan halaman <i>input</i> data mutasi pegawai
3. Pegawai memasukkan data mutasi pegawai	
	4. Sistem menambah data mutasi pegawai

**Tabel 3.23** Skenario *Use Case Login* sebagai Kepala Dinas

Aksi Aktor	Respon Sistem
1. Kadis memasukkan <i>username</i> dan <i>password</i>	
	2. Sistem mengecek <i>username</i> dan <i>password</i> , jika <i>username</i> dan <i>password</i> salah
3. <i>Admin</i> memasukkan <i>username</i> dan <i>password</i> Kembali	
	4. Sistem mengecek <i>username</i> dan <i>password</i> , jika <i>username</i> dan <i>password</i> benar
	5. Sistem menampilkan <i>menu</i> utama Kadis

**Tabel 3.24** Skenario *Use Case Monitoring Data*

Aksi Aktor	Respon Sistem
1. Kadis mengakses <i>menu monitoring data</i>	
	2. Sistem menampilkan halaman <i>monitoring data</i>
3. Kadis memilih dan mencari data	
	4. Sistem menampilkan data yang dicari dan terpilih

**Tabel 3.25** Skenario *Use Case Verifikasi Berkas*

Aksi Aktor	Respon Sistem
1. Kadis mengakses <i>menu verifikasi berkas</i>	
	2. Sistem menampilkan halaman verifikasi berkas
3. Kadis memilih berkas yang ingin diverifikasi berkas	
	4. Sistem menyimpan data yang telah terverifikasi

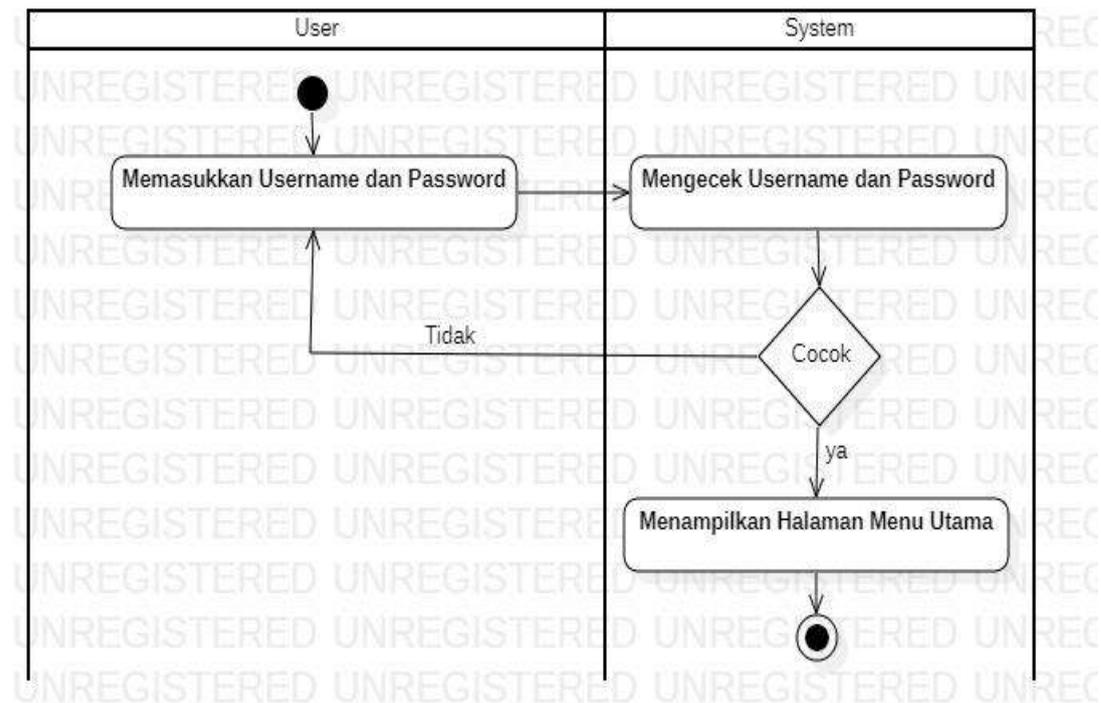
### 3.4.2 Activity Diagram

*Activity Diagram* merupakan *diagram* yang menggambarkan berbagai aliran aktivitas yang terjadi pada sistem, yang meliputi gambaran sebuah proses dapat terjadi, kemungkinan-kemungkinan yang bisa terjadi, serta hasil akhir dari proses

tersebut. Berikut ini adalah *activity diagram* dari perancangan Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta, dengan penjelasan sebagai berikut :

### 3.4.2.1 Activity Diagram Login User

*Activity Diagram Login Pegawai* dapat dilihat pada Gambar 3.3

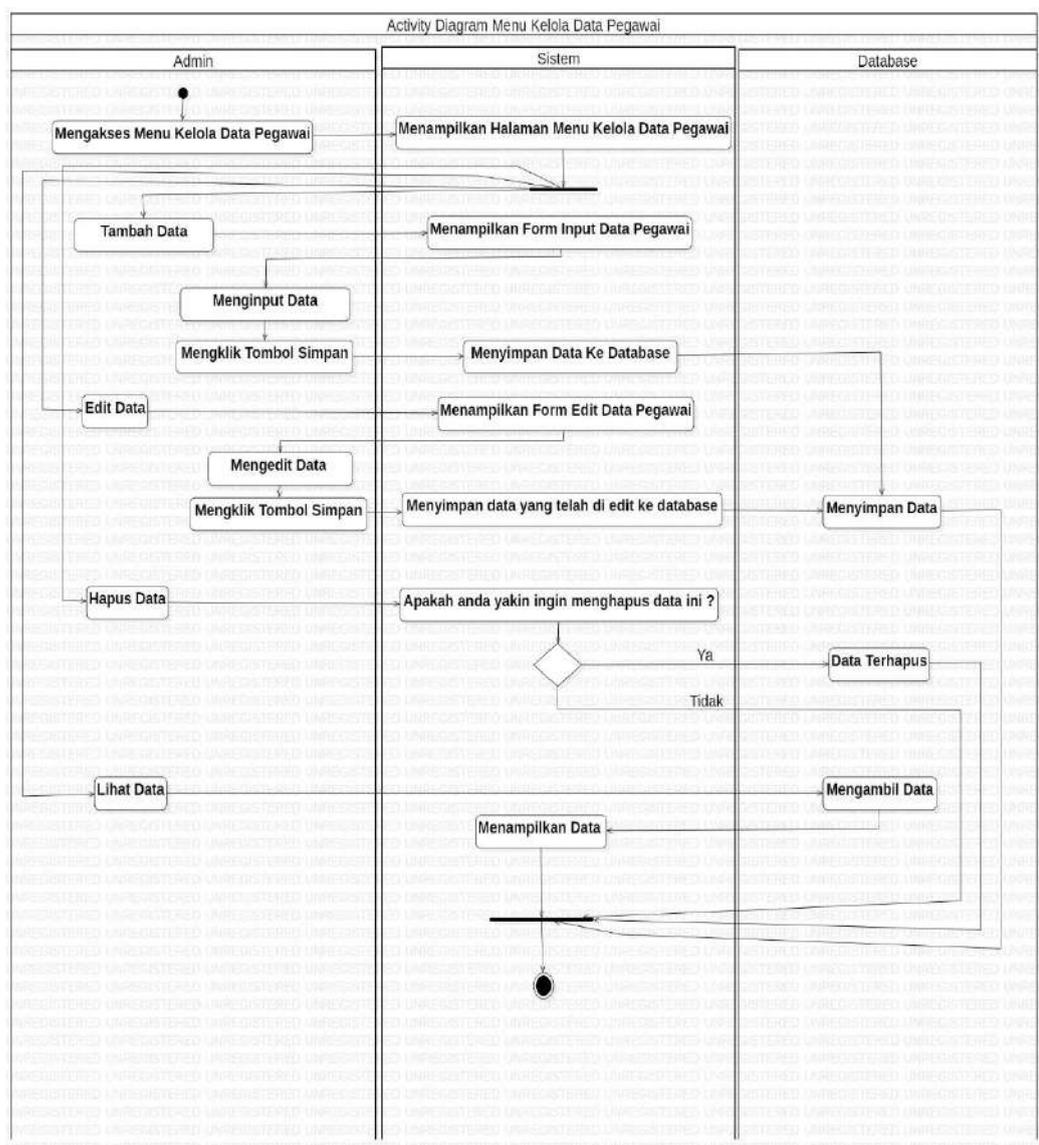


**Gambar 3.3** Activity Diagram Login User

Aktor pegawai pada *activity diagram* ini mengakses halaman *login user*, selanjutnya sistem menampilkan halaman *login user*, setelah sistem menampilkan halaman *login* maka selanjutnya pegawai memasukkan *username* dan *password*, tahapan berikutnya sistem mengecek *username* dan *password* yang telah dimasukkan oleh *user* sesuai dengan *database login* atau tidak. Jika *username* dan *password* benar maka sistem menampilkan *menu utama* dari sistem ini, sedangkan jika *username* dan *password* salah maka sistem meminta pegawai untuk memasukkan *username* dan *password* kembali.

### 3.4.2.2 Activity Diagram Kelola Data Pegawai

Activity Diagram Kelola Data Pegawai disajikan pada Gambar 3.4

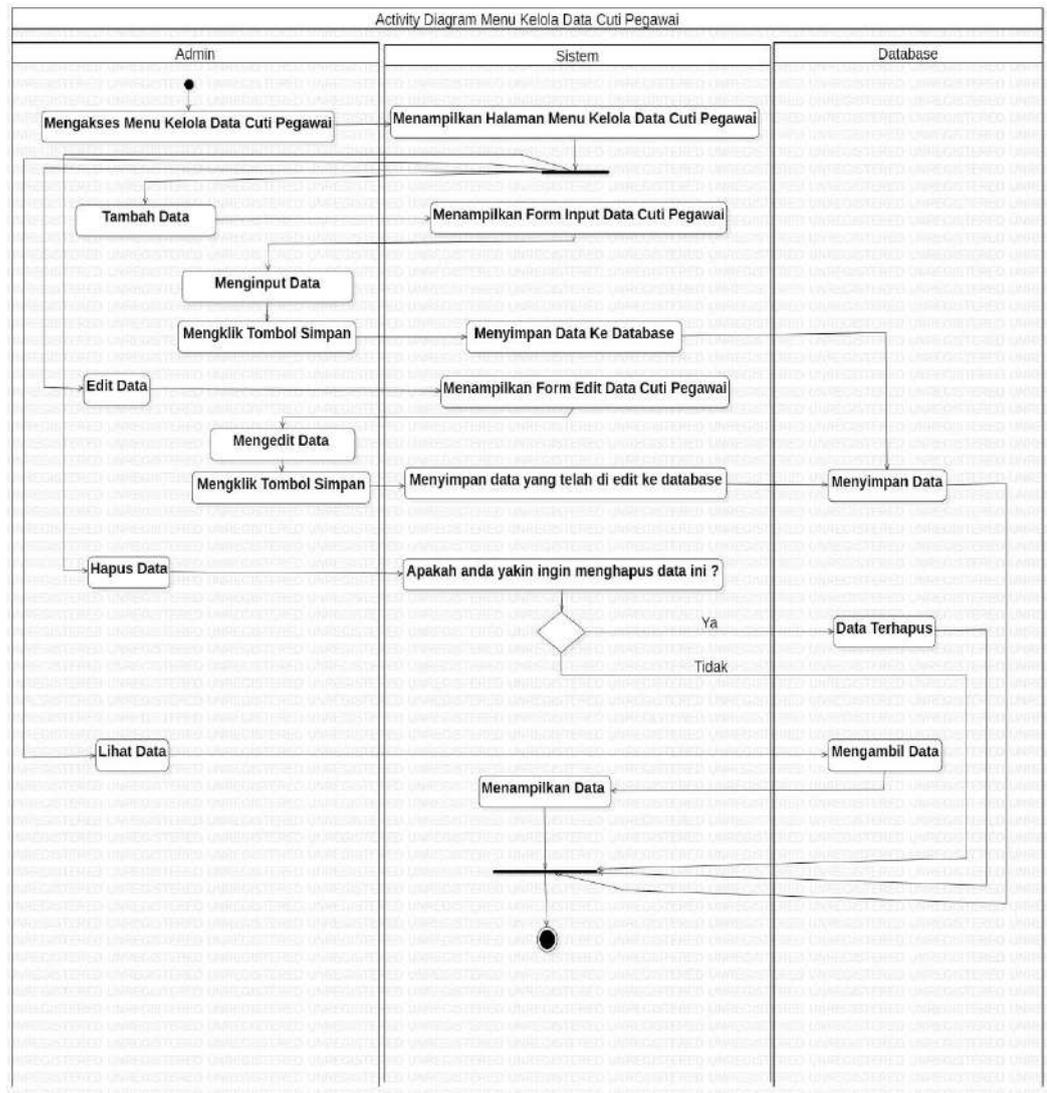


**Gambar 3.4** Activity Diagram Kelola Data Pegawai

Peran aktor *admin* pada *activity diagram* ini dapat mengakses halaman kelola data pegawai dan sistem menampilkan halaman data pegawai. *Admin* dapat mengelola data pegawai yakni *admin* dapat menambah, mengedit, menghapus serta menyimpan data pegawai. Setelah sistem menyimpan data, apabila *admin* tidak ingin memperbaharui data maka *activity diagram* berakhir.

### 3.4.2.3 Activity Diagram Kelola Data Cuti Pegawai

Activity Diagram Kelola Data Cuti Pegawai dapat dilihat pada Gambar 3.5

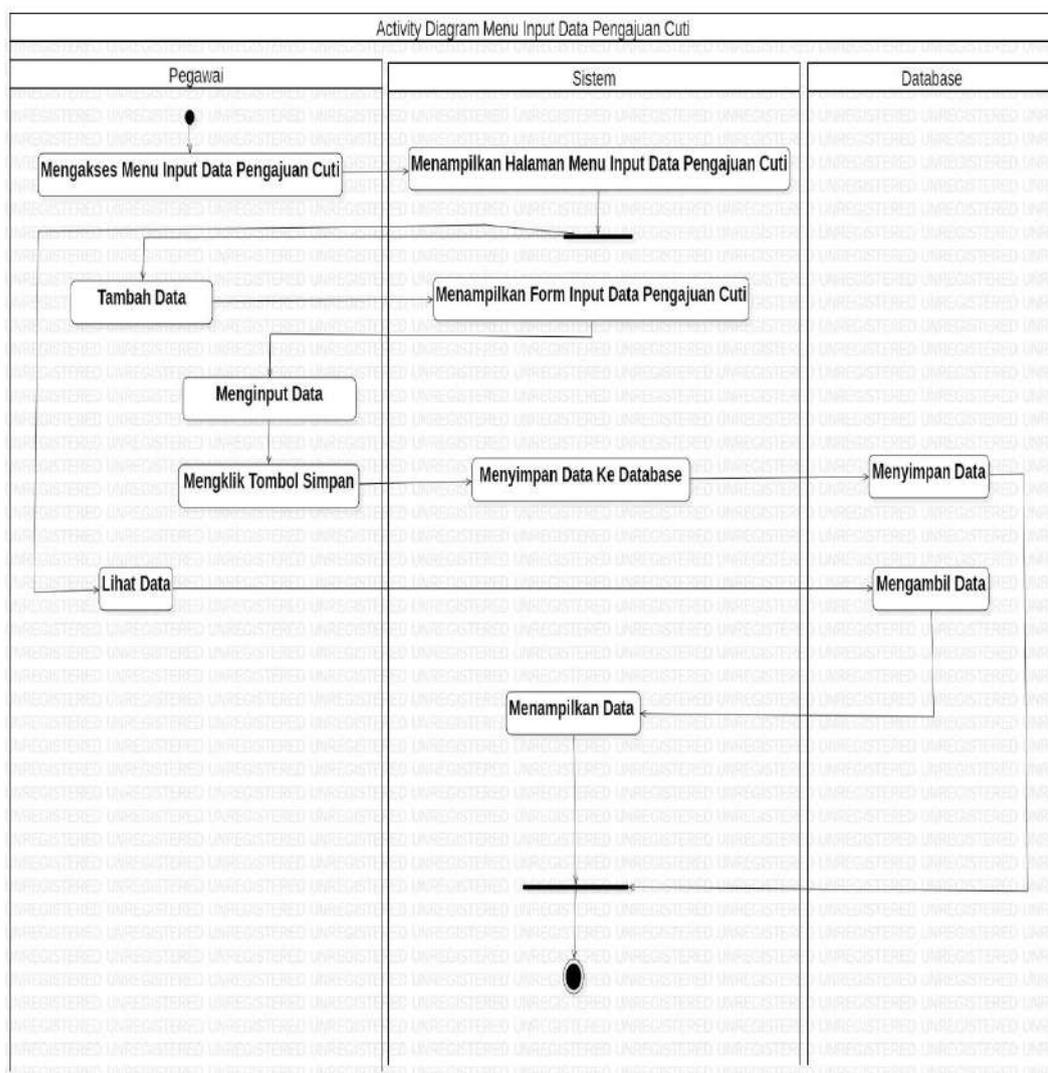


**Gambar 3.5** Activity Diagram Kelola Data Cuti Pegawai

Admin dalam activity diagram ini mengakses halaman kelola data cuti pegawai, selanjutnya sistem menampilkan halaman data cuti pegawai. Tahapan berikutnya yaitu admin mengelola data cuti pegawai yang dimana admin dapat menambah, mengedit, menghapus serta menyimpan data cuti pegawai. Setelah itu sistem menyimpan data, apabila aktor admin tidak ingin memperbaharui data maka activity diagram berakhir.

### 3.4.2.4 Activity Diagram Input Data Pengajuan Cuti

Activity Diagram Input Data Pengajuan Cuti disajikan pada Gambar 3.6

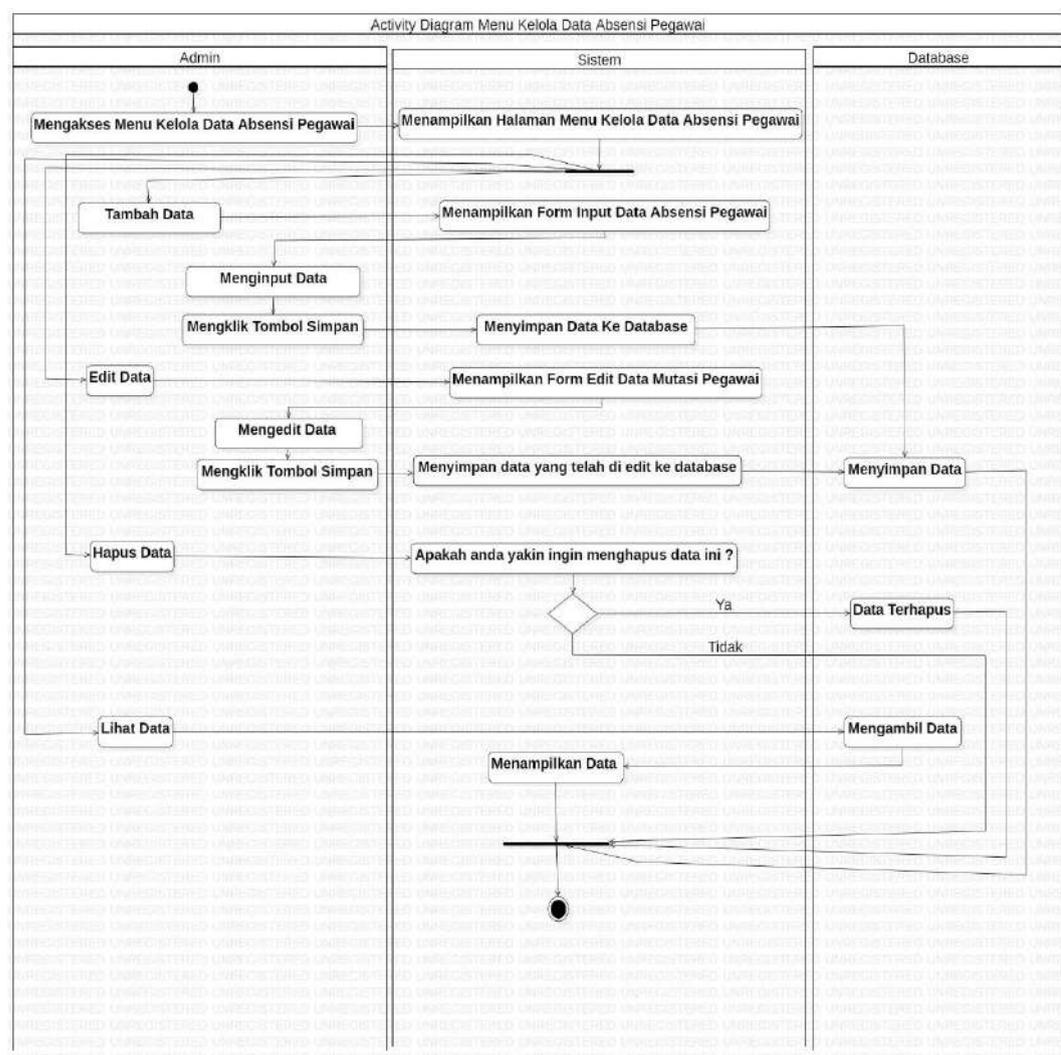


**Gambar 3.6** Activity Diagram Input Data Pengajuan Cuti

Aktor pegawai dalam *activity diagram* ini melakukan tahapan pertama dengan memilih *menu input* data pengajuan cuti, selanjutnya sistem menampilkan halaman *input* data pengajuan cuti. Tahapan berikutnya yakni pegawai dapat mengisi *form* data pengajuan cuti yang sudah tersedia, setelah mengisi *form* pengajuan cuti yang sudah ada kemudian pegawai menekan tombol simpan, maka sistem menyimpan data pengajuan cuti yang diajukan oleh pegawai.

### 3.4.2.5 Activity Diagram Kelola Data Absensi Pegawai

Activity Diagram Kelola Data Absensi Pegawai disajikan pada Gambar 3.7

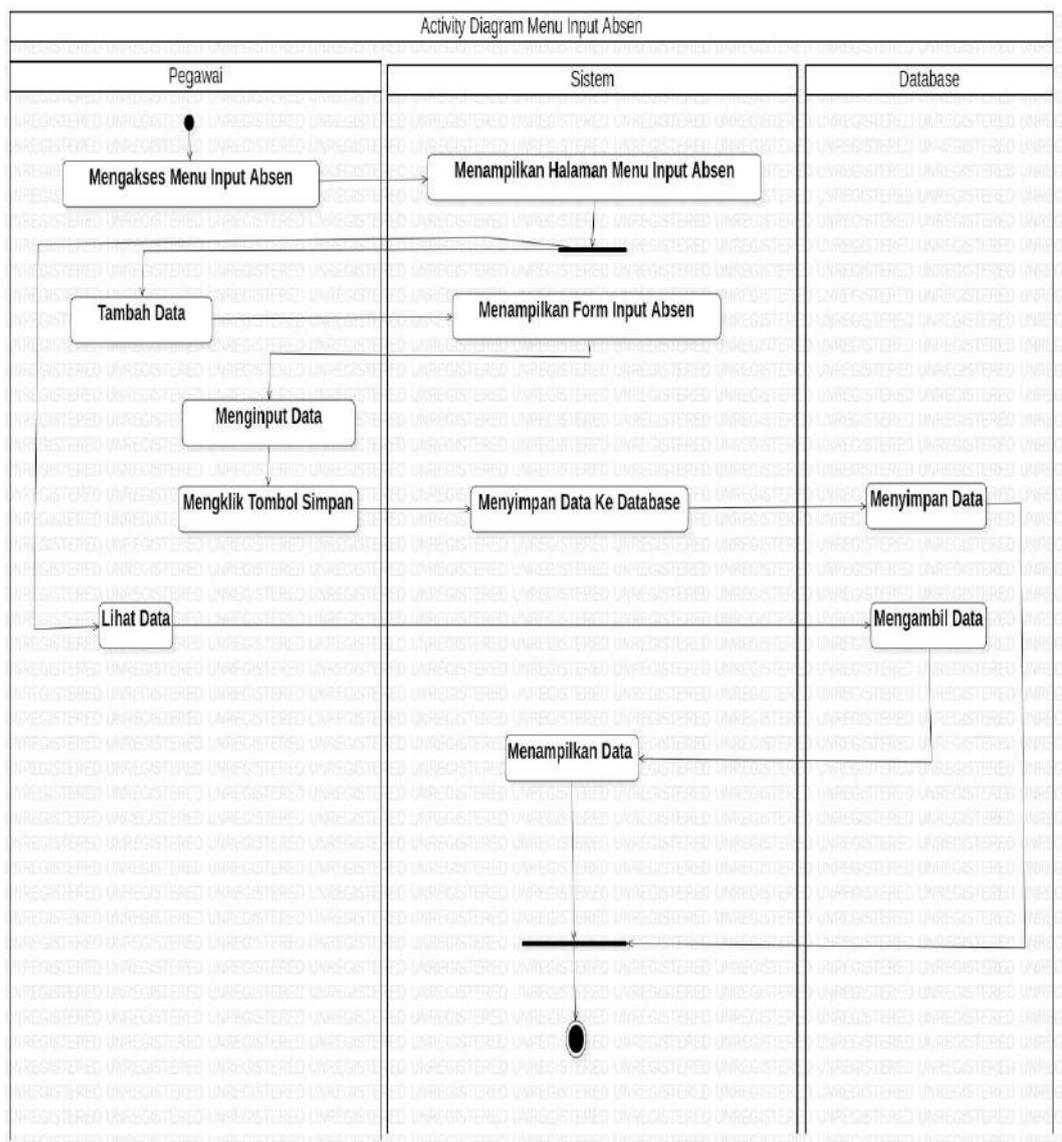


**Gambar 3.7** Activity Diagram Kelola Data Absensi Pegawai

Aktor sebagai *admin* dalam *activity diagram* ini, admin dapat mengakses halaman kelola data absensi pegawai, berikutnya sistem menampilkan halaman data absensi pegawai. Tahapan selanjutnya *admin* mengelola data absensi pegawai yakni *admin* dapat menambah, mengedit, menghapus dan menyimpan data absensi pegawai. Setelah itu sistem menyimpan data dan apabila *admin* tidak ingin memperbaharui data maka *activity diagram* berakhir.

### 3.4.2.6 Activity Diagram Input Absen

Activity Diagram Input Absen dapat diperhatikan pada Gambar 3.8

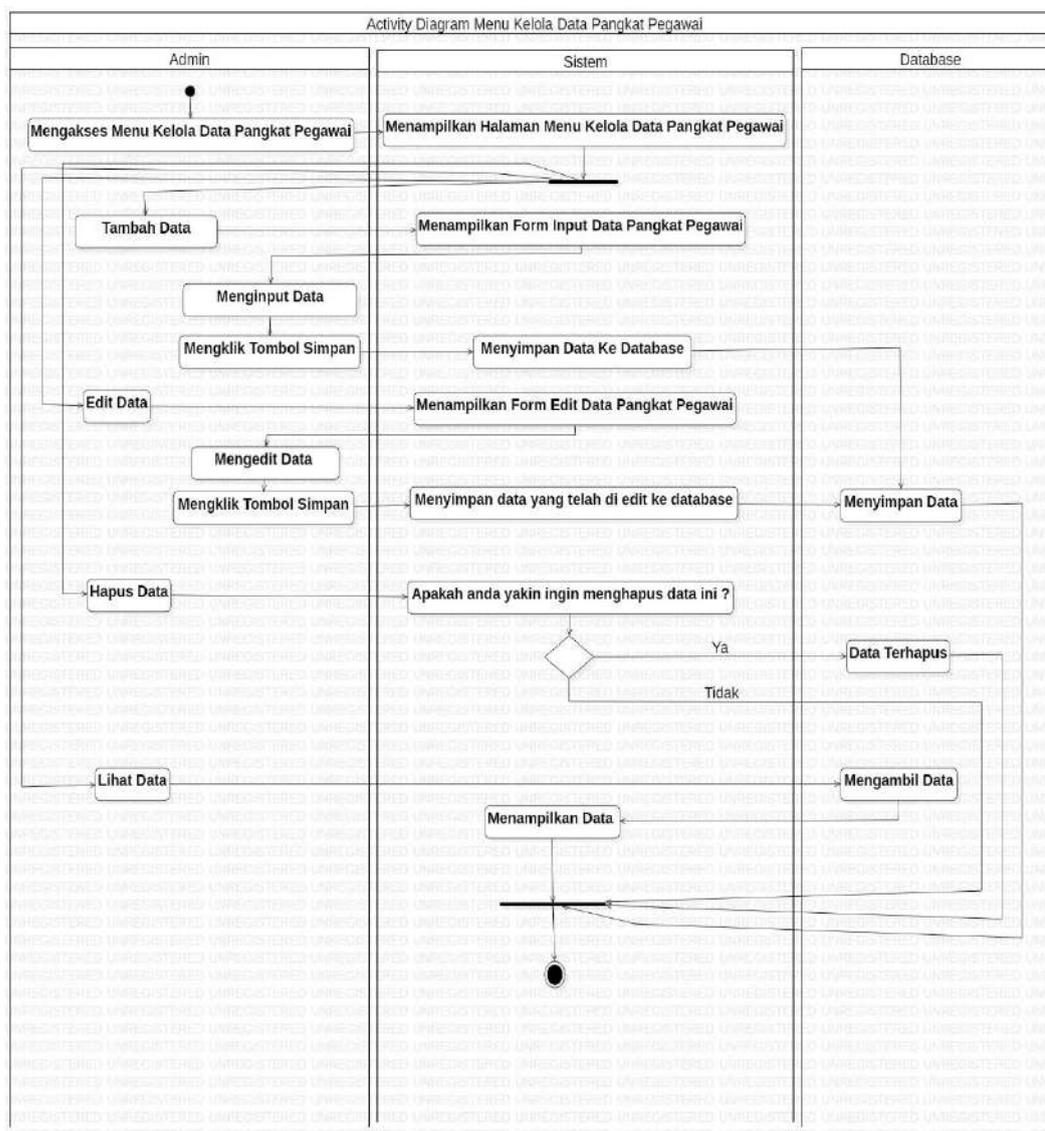


**Gambar 3.8** Activity Diagram Input Absen

Aktor pegawai dalam *activity diagram* ini memilih *menu input* absen, selanjutnya sistem menampilkan halaman *input* absen yang sudah tersedia. Tahap berikutnya pegawai mengisi *form* absen untuk keperluan izin ataupun sakit beserta dengan keterangan dan bukti yang akurat, setelah pegawai mengisi *form* absen, maka sistem menyimpan absen secara otomatis.

### 3.4.2.7 Activity Diagram Kelola Data Pangkat Pegawai

Activity Diagram Kelola Data Pangkat Pegawai dapat dilihat pada Gambar 3.9

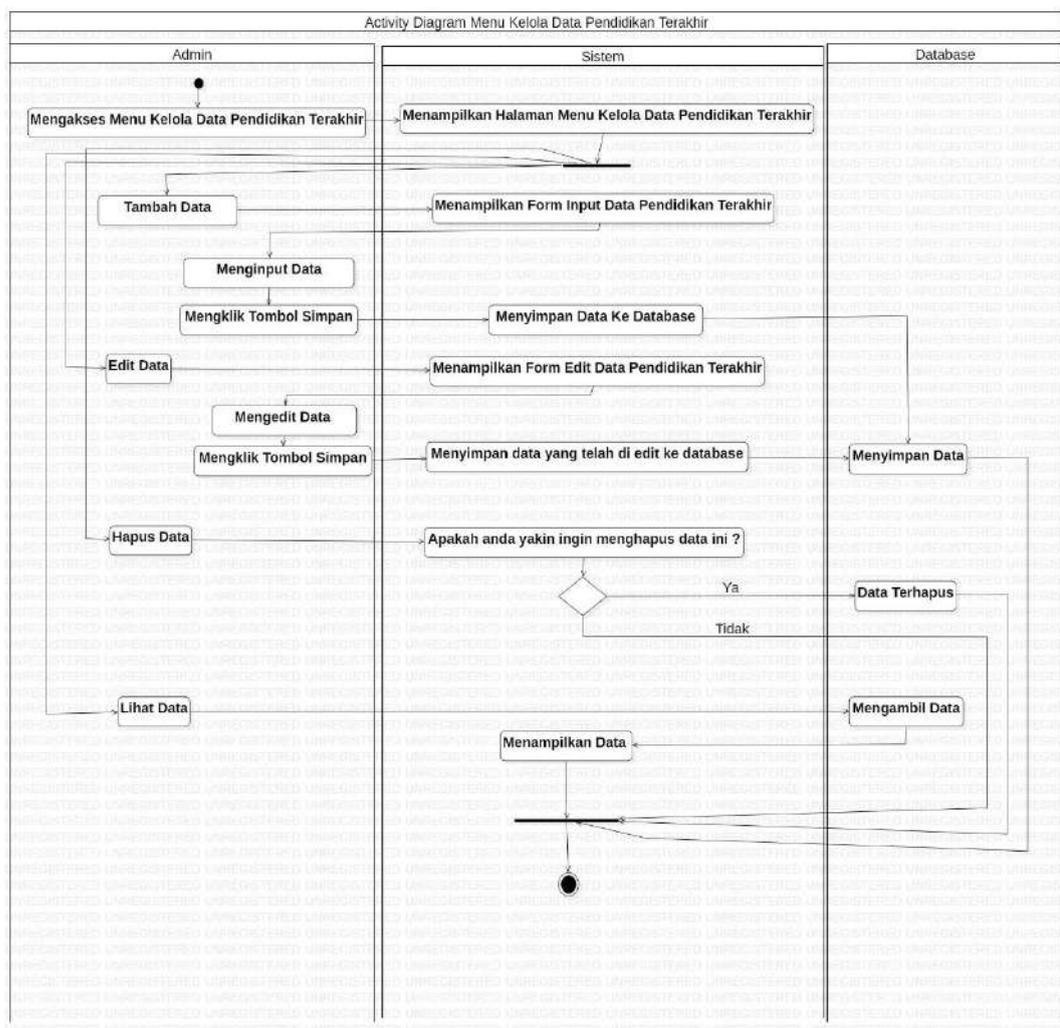


**Gambar 3.9** Activity Diagram Kelola Data Pangkat Pegawai

Admin dalam activity diagram ini dapat mengakses halaman kelola data pangkat pegawai, maka sistem menampilkan halaman data pangkat pegawai. Selanjutnya admin bertugas untuk mengelola data pangkat pegawai yakni admin dapat menambah, mengedit, menghapus, menyimpan data pangkat pegawai. Setelah itu sistem menyimpan data dan apabila admin tidak ingin memperbaharui data maka activity diagram berakhir.

### 3.4.2.8 Activity Diagram Kelola Data Pendidikan Terakhir

Activity Diagram Kelola Data Pendidikan Terakhir disajikan pada Gambar 3.10

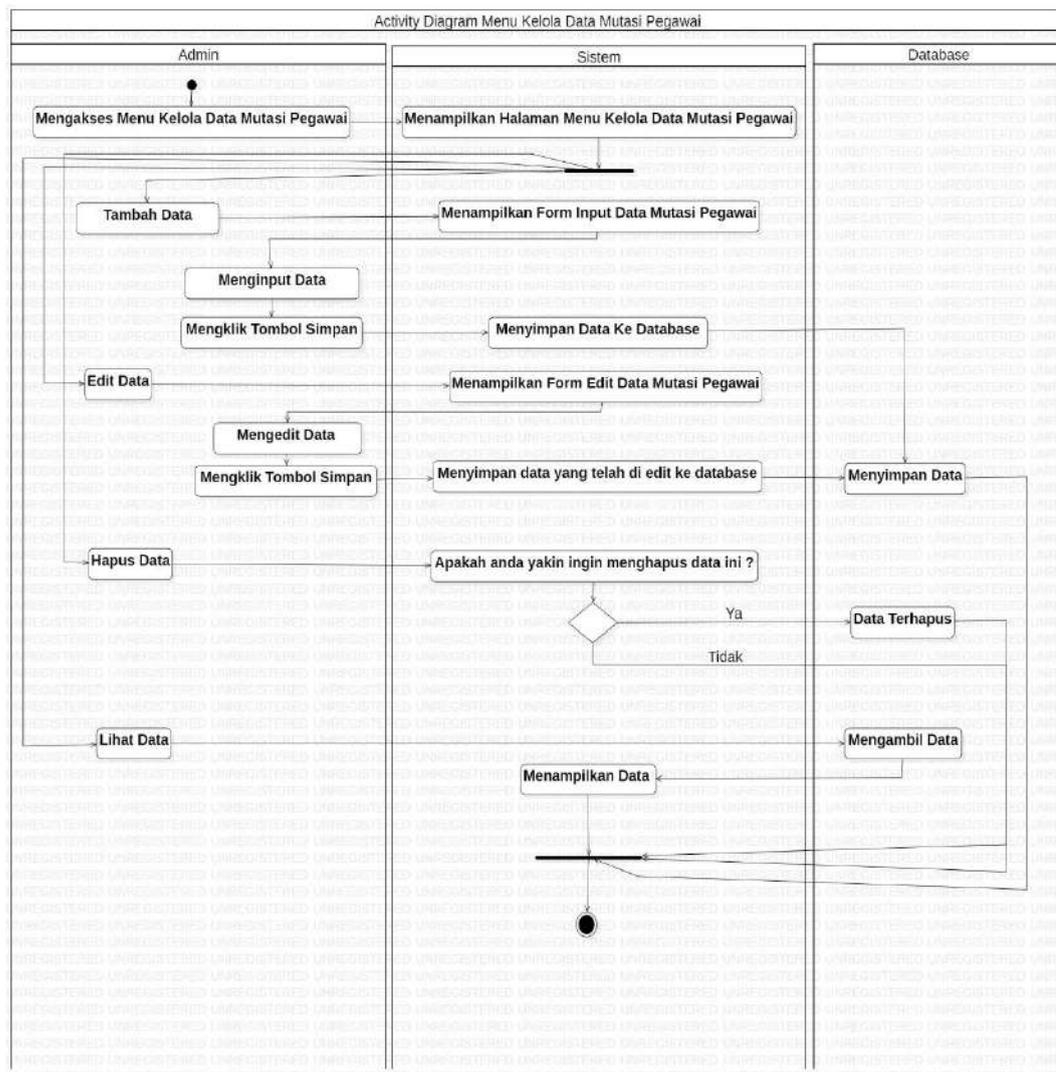


**Gambar 3.10** Activity Diagram Kelola Data Pendidikan Terakhir

Aktor atau seorang *admin* dalam *activity diagram* ini dapat mengakses halaman kelola data pendidikan terakhir, setelah itu sistem menampilkan halaman data pendidikan terakhir. Tahapan berikutnya *admin* mengelola data pendidikan terakhir yakni *admin* dapat menambah, mengedit, menghapus dan menyimpan data pendidikan terakhir. Setelah dilakukannya pembaharuan data, maka sistem menyimpan data secara otomatis, apabila *admin* tidak ingin memperbaharui data maka *activity diagram* berakhir.

### 3.4.2.9 Activity Diagram Kelola Data Mutasi Pegawai

Activity Diagram Kelola Data Mutasi Pegawai dapat dilihat pada Gambar 3.11

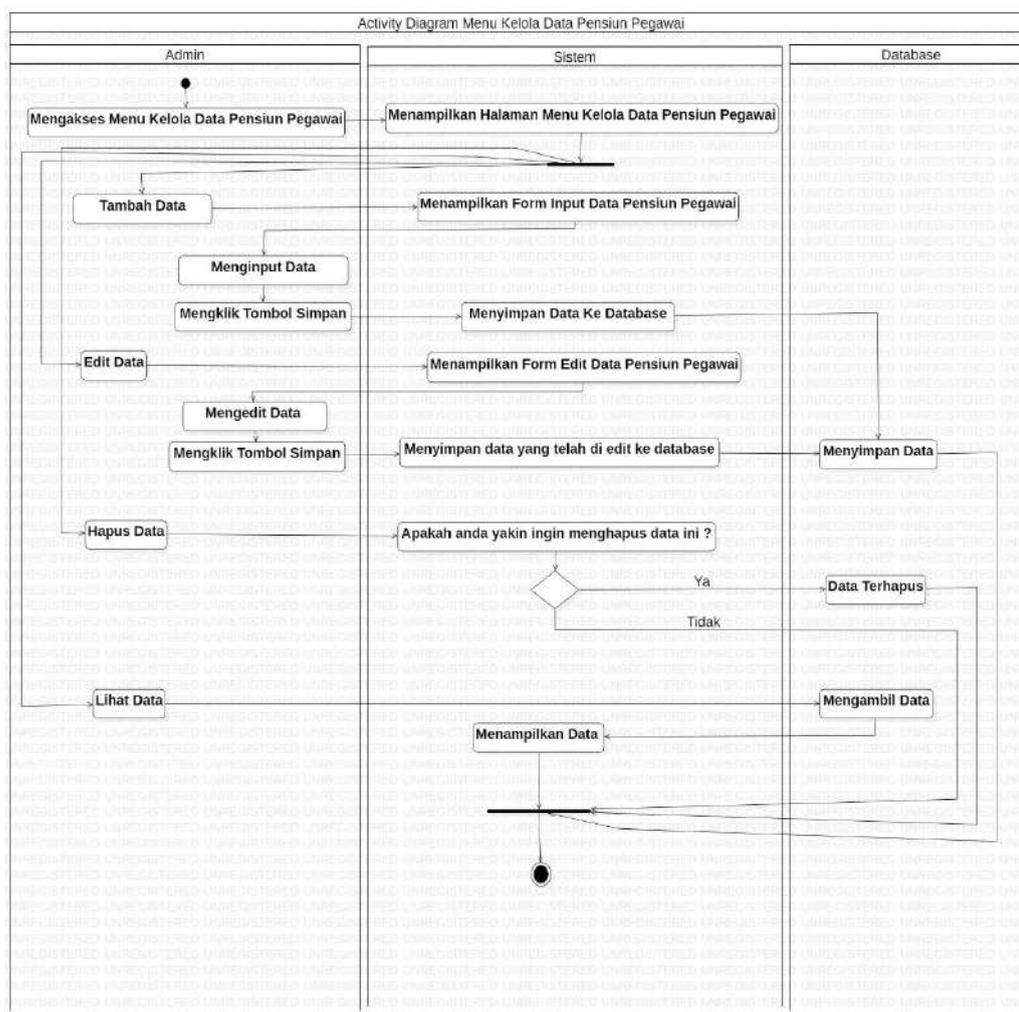


**Gambar 3.11** Activity Diagram Kelola Data Mutasi Pegawai

Aktor *admin* dalam *activity diagram* ini mengakses halaman kelola data mutasi pegawai, setelah *admin* mengakses halaman kelola data mutasi pegawai, maka sistem menampilkan halaman data mutasi pegawai. Selanjutnya *admin* mengelola data mutasi pegawai yakni dimana *admin* dapat menambah, mengedit, menghapus dan menyimpan data mutasi pegawai. Berikutnya sistem menyimpan data dan apabila *admin* tidak ingin memperbaharui data maka *activity diagram* berakhir.

### 3.4.2.10 Activity Diagram Kelola Data Pensiun Pegawai

Activity Diagram Kelola Data Pensiun Pegawai disajikan pada Gambar 3.12

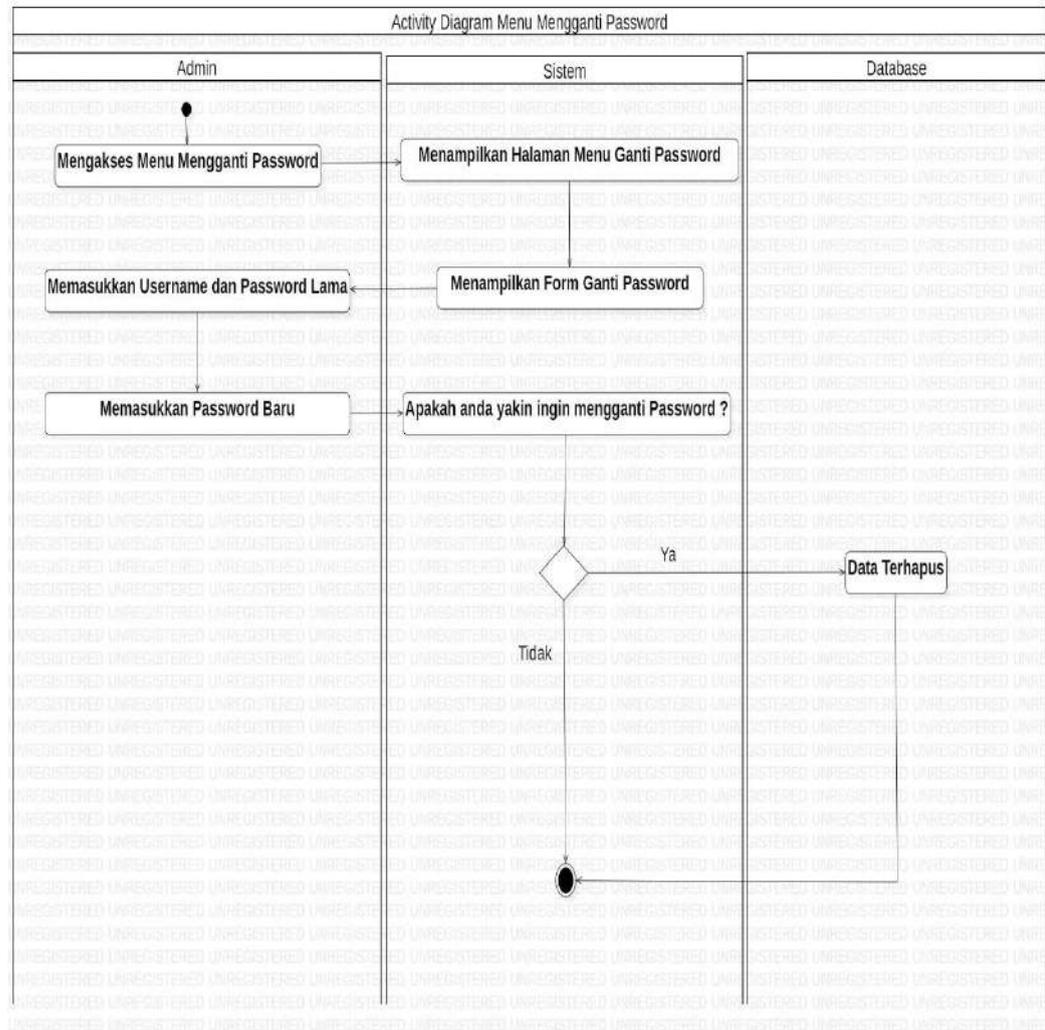


**Gambar 3.12** Activity Diagram Kelola Data Pensiun Pegawai

Aktor atau seorang *admin* dalam *activity diagram* ini dengan tahapan pertama yakni mengakses halaman kelola data pensiun pegawai, setelah itu maka sistem menampilkan halaman data pensiun pegawai. Selanjutn *admin* mengelola data pensiun pegawai yakni dimana *admin* dapat menambah, mengedit, menghapus dan menyimpan data pensiun pegawai. Setelah dilakukannya pembaharuan data sistem menyimpan data dan apabila *admin* tidak ingin memperbaharui data maka *activity diagram* berakhir.

### 3.4.2.11 Activity Diagram Mengganti Password

Activity Diagram Mengganti Password dapat diperhatikan pada Gambar 3.13

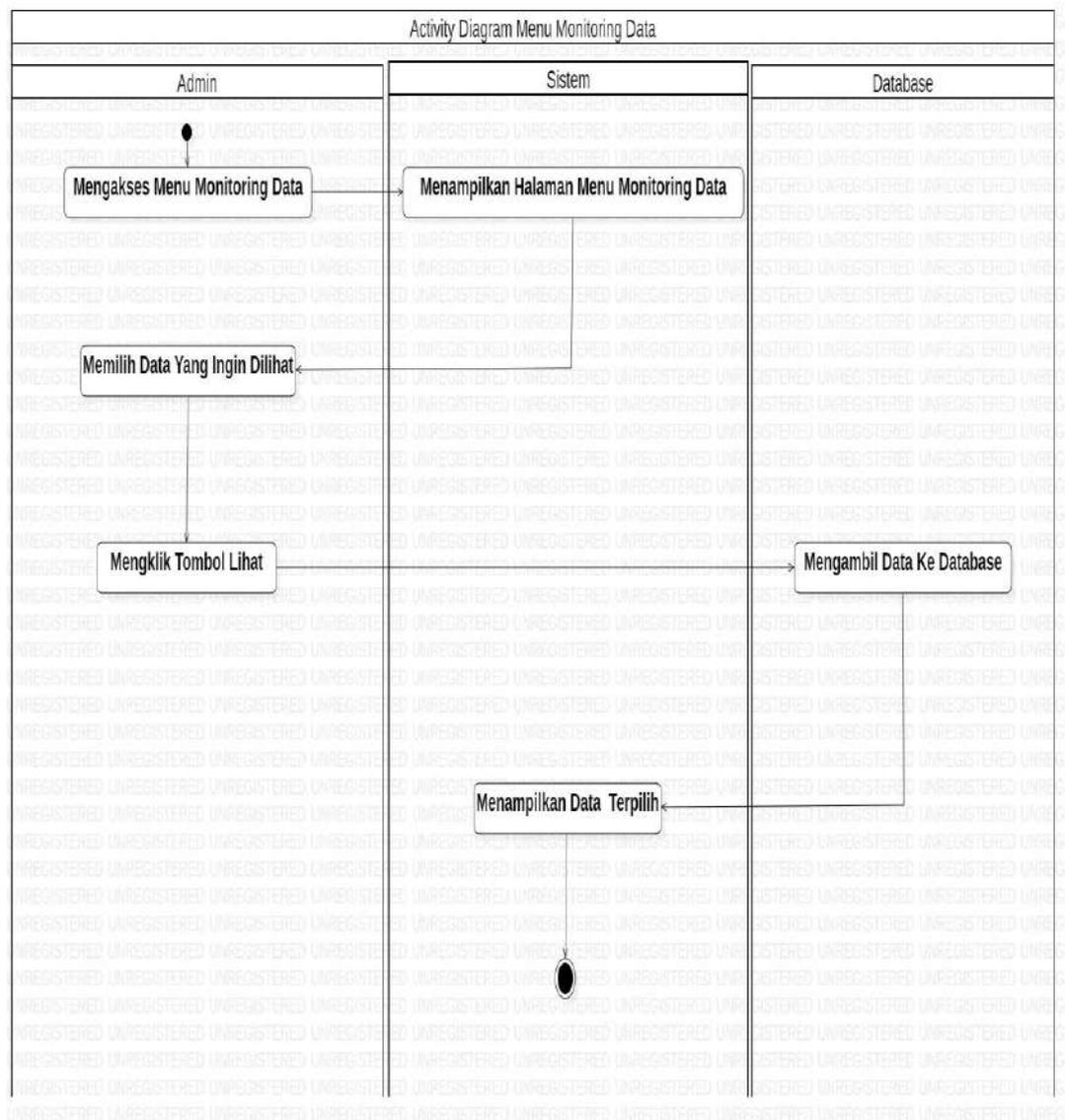


**Gambar 3.13** Activity Diagram Mengganti Password

Aktor yang berperan sebagai *admin* dalam *activity diagram* ini langkah pertama dengan mengakses halaman mengganti *password*, selanjutnya sistem menampilkan halaman mengganti *password*. Tahapan berikutnya seorang *admin* memasukkan *username* dan *password* yang diganti selanjutnya masukkan *password* lama, kemudian *password* baru dan konfirmasi *password* baru. Setelah melakukan beberapa tahapan diatas maka sistem mengkonfirmasi *password* baru secara otomatis dan menyimpan *password* yang telah diganti.

### 3.4.2.12 Activity Diagram Monitoring Data

Activity Diagram Monitoring Data disajikan pada Gambar 3.14

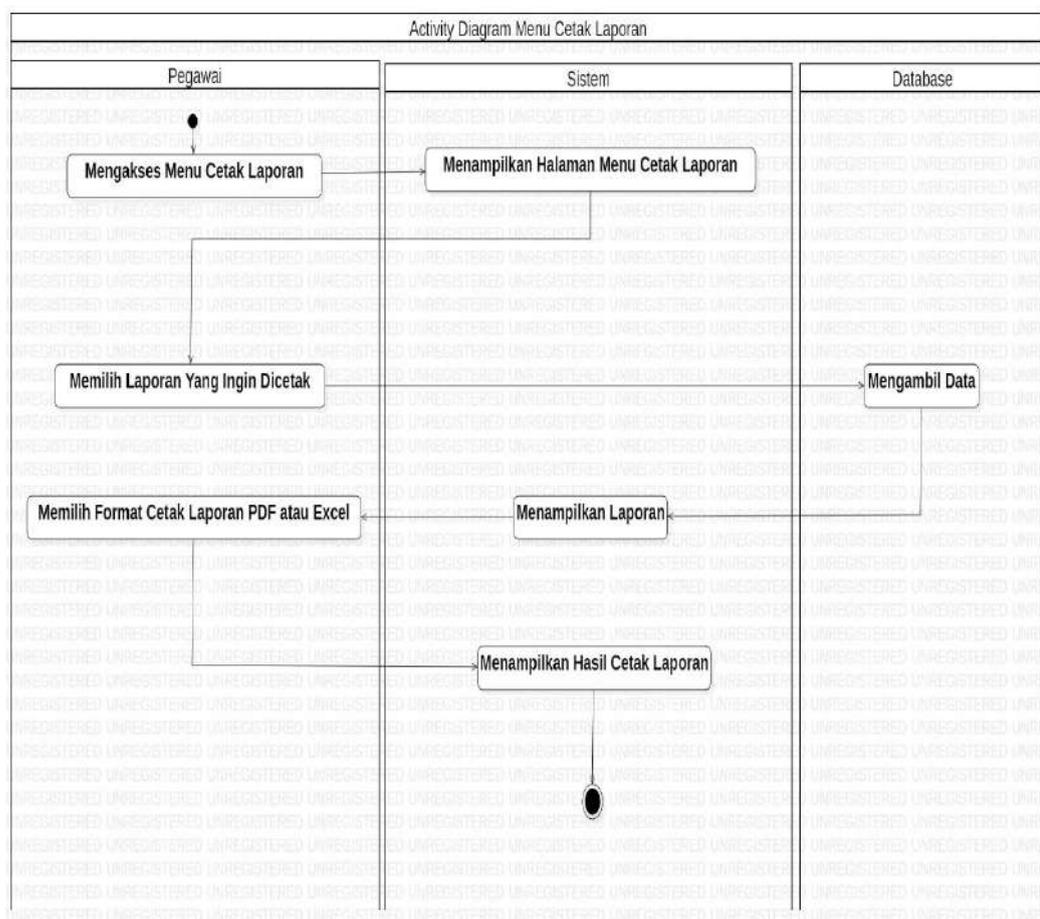


**Gambar 3.14** Activity Diagram Monitoring Data

*Admin* dalam *activity diagram* diatas dapat mengakses halaman *monitoring data*, setelah itu sistem menampilkan halaman *monitoring data*. Selanjutnya *admin* dapat mengecek data pegawai yang masih belum lengkap data dari pegawai.

### 3.4.2.13 Activity Diagram Cetak Laporan

Activity Diagram Cetak Laporan dapat dilihat pada Gambar 3.15

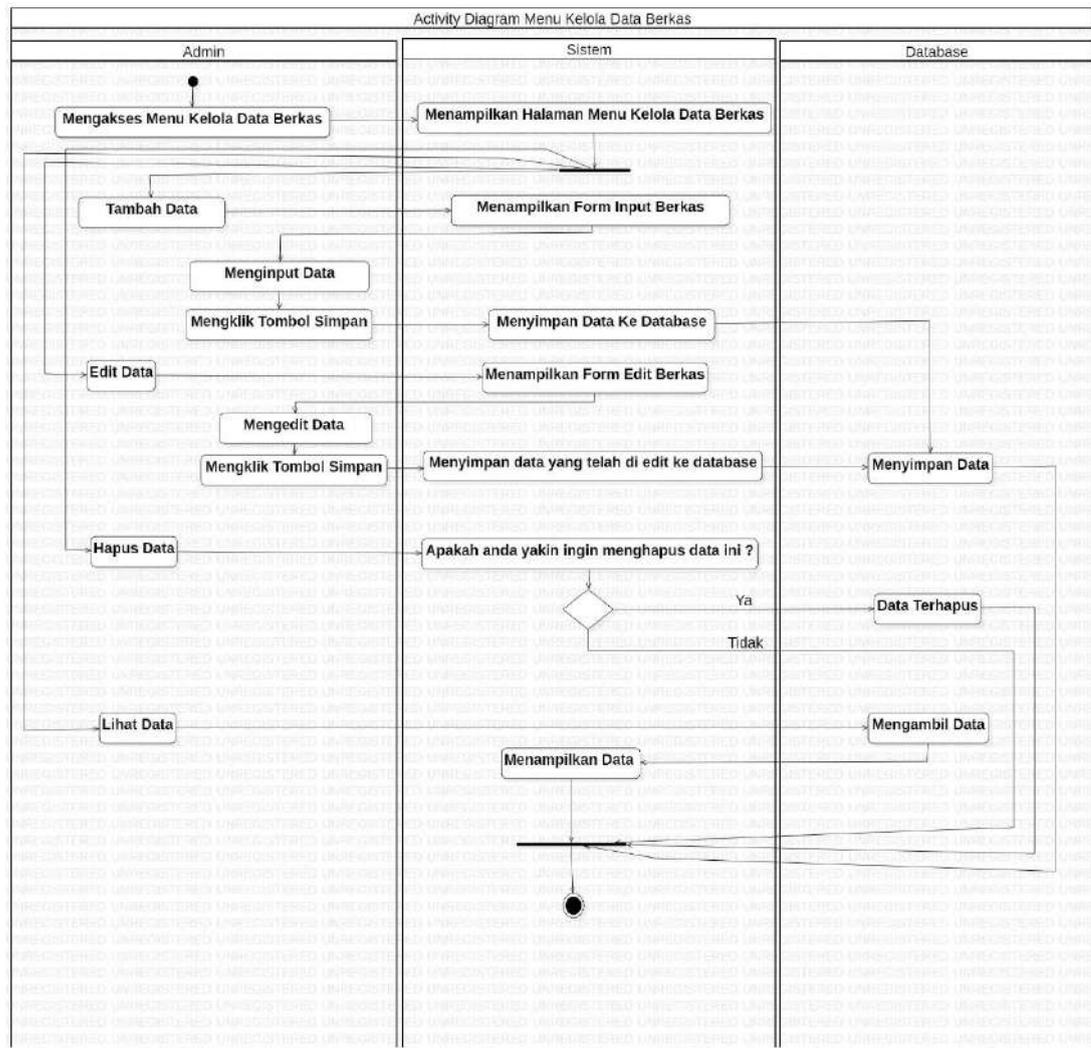


**Gambar 3.15** Activity Diagram Cetak Laporan

Aktor atau seorang *admin* dalam *activity diagram* diatas dapat mengakses halaman cetak laporan, selanjutnya sistem menampilkan halaman cetak laporan. Setelah sistem menampilkan halaman cetak laporan, maka *admin* dapat memilih masing-masing data yang ingin dicetak, selanjutnya *admin* memilih data berdasarkan tahun, dan memilih data berdasarkan bulan yang ingin dicetak. Tahapan berikutnya *admin* memilih perintah *print* atau *ekspor* data dalam bentuk *pdf* atau *microsoft excel*. Setelah melakukan tahapan diatas sistem menampilkan hasil laporan yang telah dicetak.

### 3.4.2.14 Activity Diagram Kelola Data Berkas

Activity Diagram Kelola Data Berkas disajikan pada Gambar 3.16

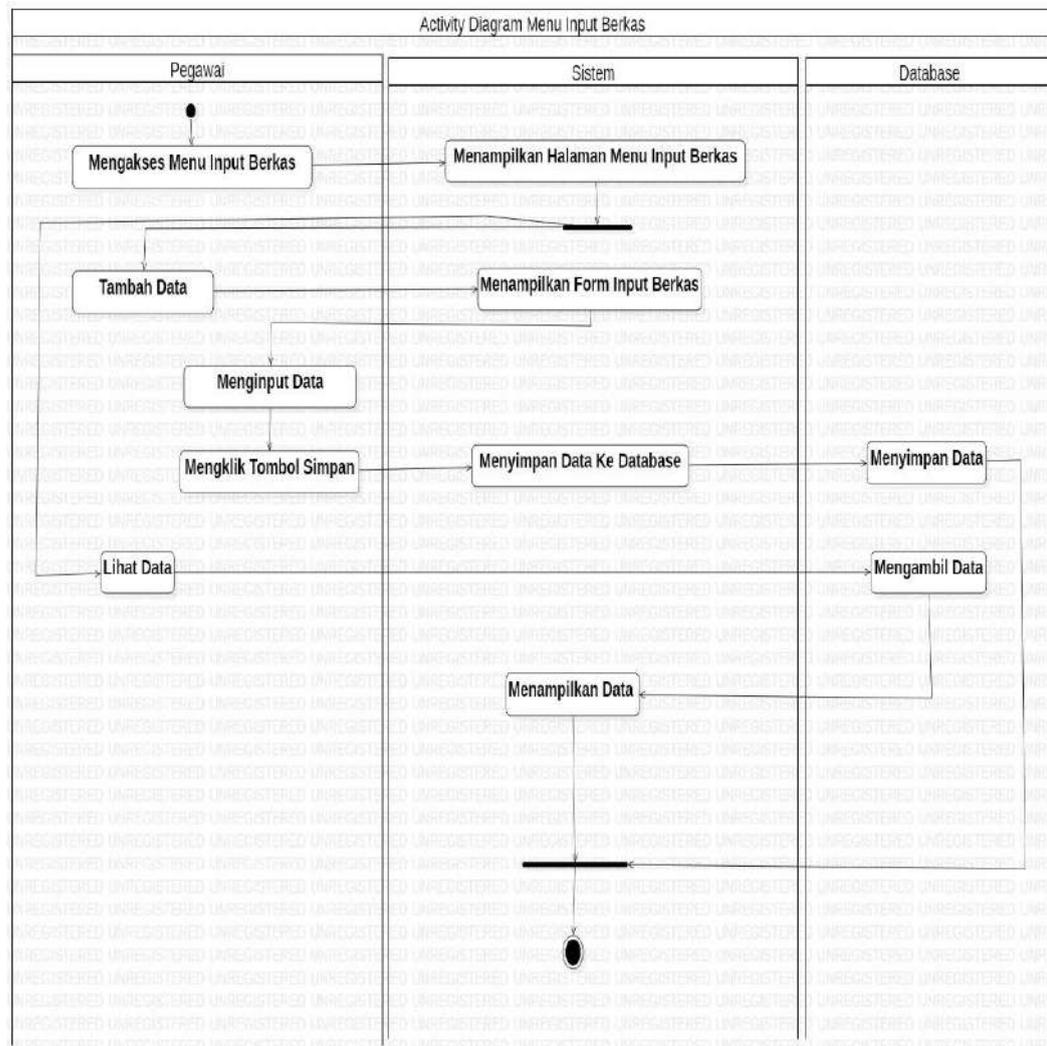


**Gambar 3.16** Activity Diagram Kelola Data Berkas

Dalam *activity diagram* ini *admin* dapat mengakses halaman kelola data berkas, selanjutnya sistem menampilkan halaman data berkas. Setelah tampil halaman *admin* mempunyai hak akses sepenuhnya untuk mengelola data berkas yang dimana *admin* dapat menambah, menghapus dan menyimpan data berkas. berikutnya sistem menyimpan data dan apabila *admin* tidak adanya pembaharuan data maka *activity diagram* berakhir.

### 3.4.2.15 Activity Diagram Input Berkas

Activity Diagram Input Berkas dapat diperhatikan Gambar 3.17

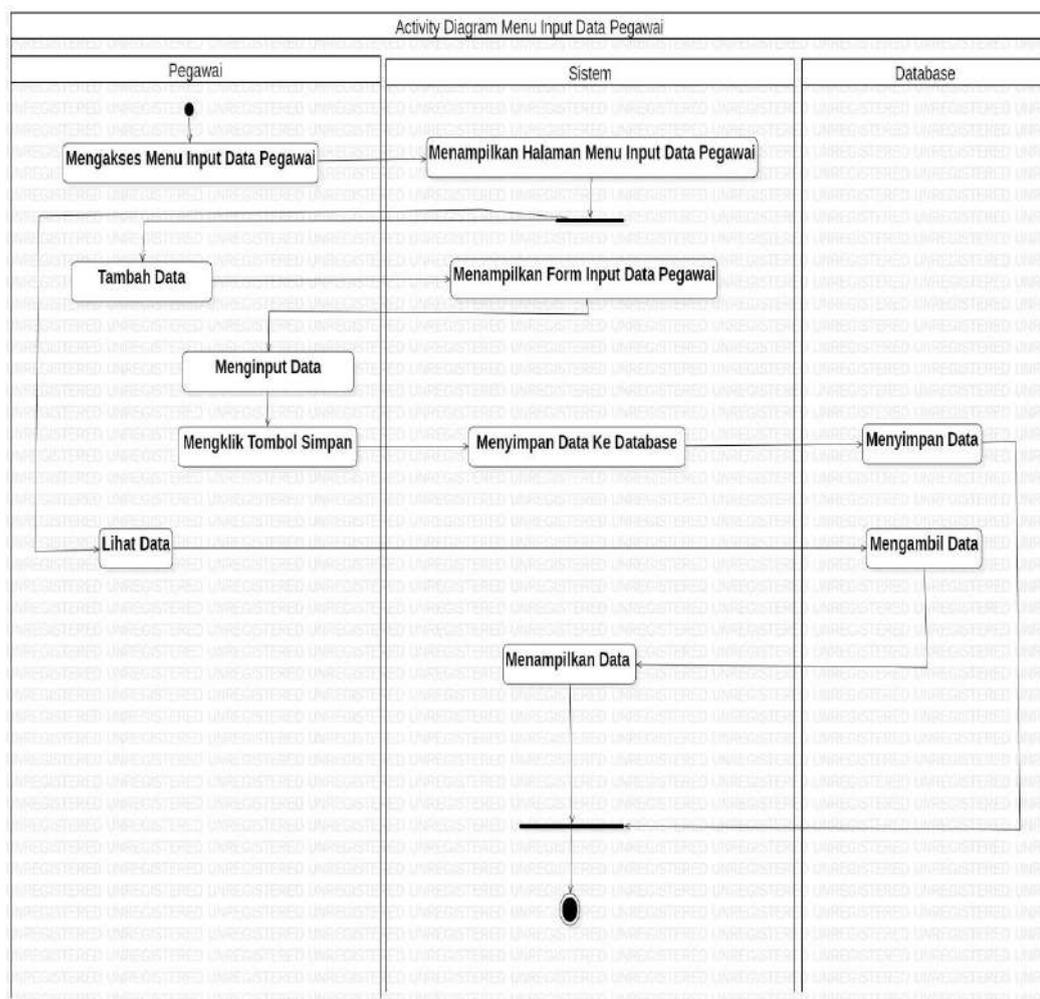


**Gambar 3.17** Activity Diagram Input Berkas

Pada *activity diagram* ini actor pegawai memilih *menu input* berkas, setelah itu sistem menampilkan halaman *input* berkas. Setelah sistem menampilkan halaman untuk input berkas maka tahapan selanjutnya adalah pegawai memasukkan berkas dengan format jpg, ataupun png, jika pegawai sudah memasukkan berkas tersebut berikutnya pegawai mengklik tombol simpan maka sistem menyimpan berkas yang telah dimasukkan oleh pegawai secara otomatis.

### 3.4.2.16 Activity Diagram Input Data Pegawai

Activity Diagram Input Data Pegawai disajikan pada Gambar 3.18

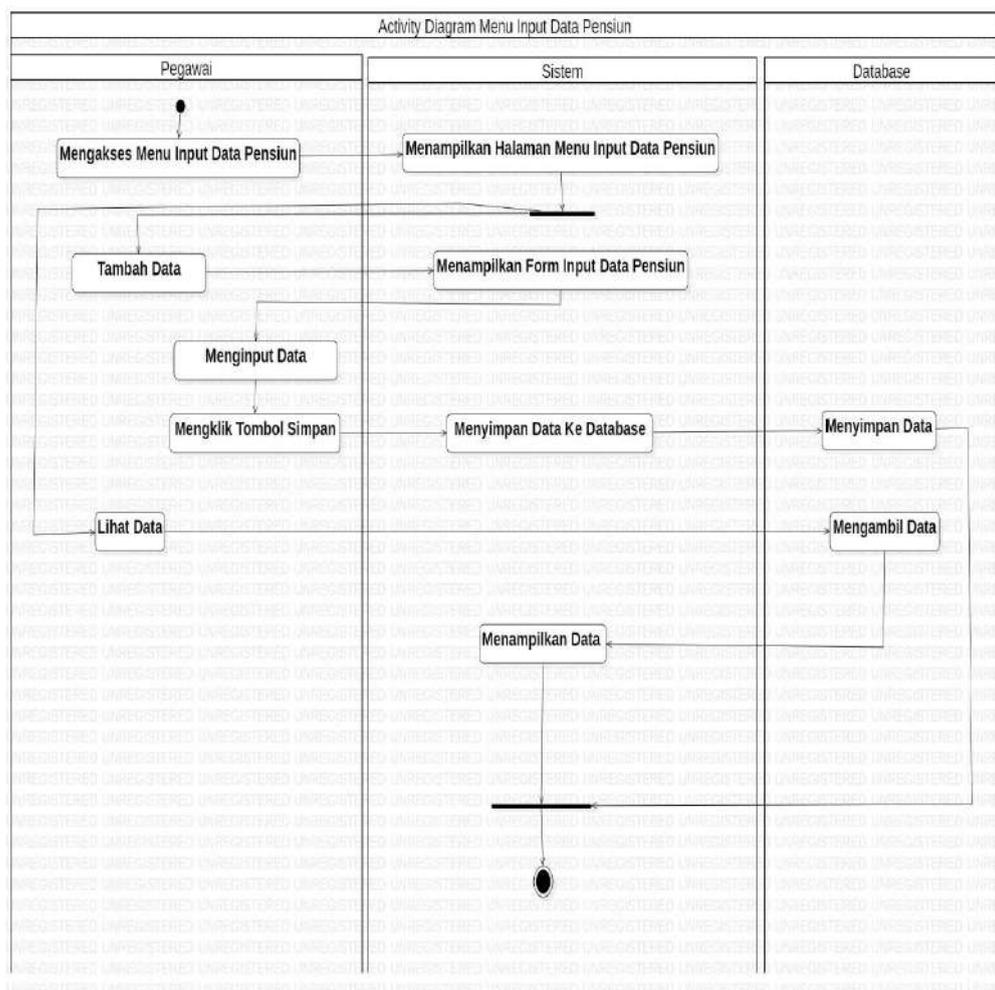


**Gambar 3.18** Activity Diagram Input Data Pegawai

Aktor pegawai yang berperan dalam *activity diagram* ini memilih *menu input data* pegawai, setelah itu sistem menampilkan halaman *input data* pegawai untuk dapat memasukkan data oleh masing-masing pegawai yang sudah tersedia di halaman *input data* pegawai. Selanjutnya pegawai dapat mengisi *form data* pegawai, setelah pegawai mengisi *form data* pegawai tahapan berikutnya adalah pegawai menekan tombol simpan, kemudian sistem menyimpan data pegawai secara otomatis yang sebelumnya sudah dimasukkan.

### 3.4.2.17 Activity Diagram Input Data Pensiun

Activity Diagram Input Data Pensiun dapat diperhatikan pada Gambar 3.19

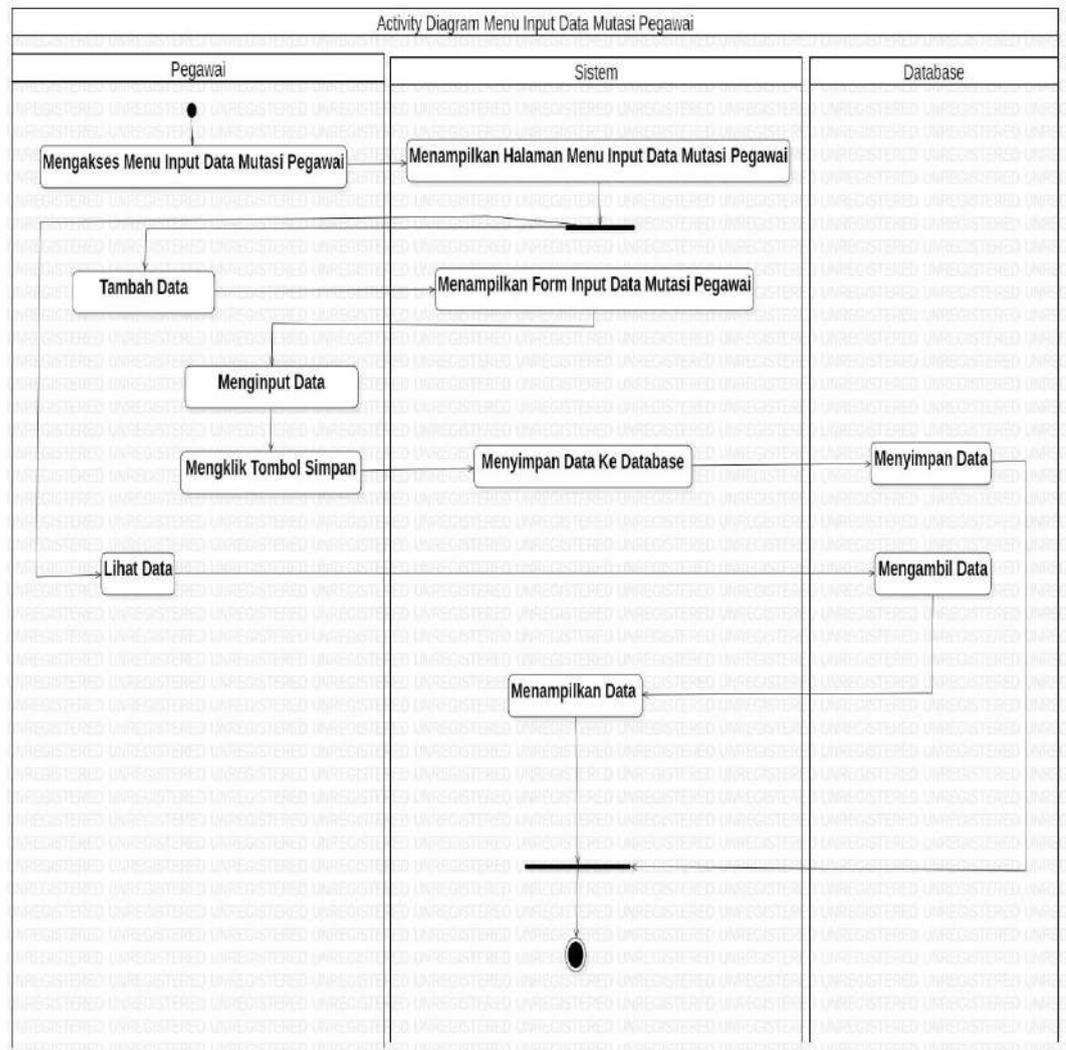


**Gambar 3.19** Activity Diagram Input Data Pensiun

Dalam *activity diagram* ini aktor atau seorang pegawai dapat memilih *menu input data pensiun*, langkah selanjutnya sistem menampilkan halaman *input data pensiun* dihalaman ini pegawai dapat mengisi data untuk pengajuan pensiun yang sudah tersedia dalam *menu* ini. Berikutnya masing-masing pegawai dapat mengisi *form data pensiun* yang sudah disediakan, jika pegawai sudah mengisi *form pengajuan pensiun*, tahapan berikutnya adalah pegawai menekan tombol simpan dengan demikian sistem menyimpan data pensiun secara otomatis.

### 3.3.2.18 Activity Diagram Input Data Mutasi Pegawai

Activity Diagram Input Data Mutasi Pegawai disajikan pada Gambar 3.20

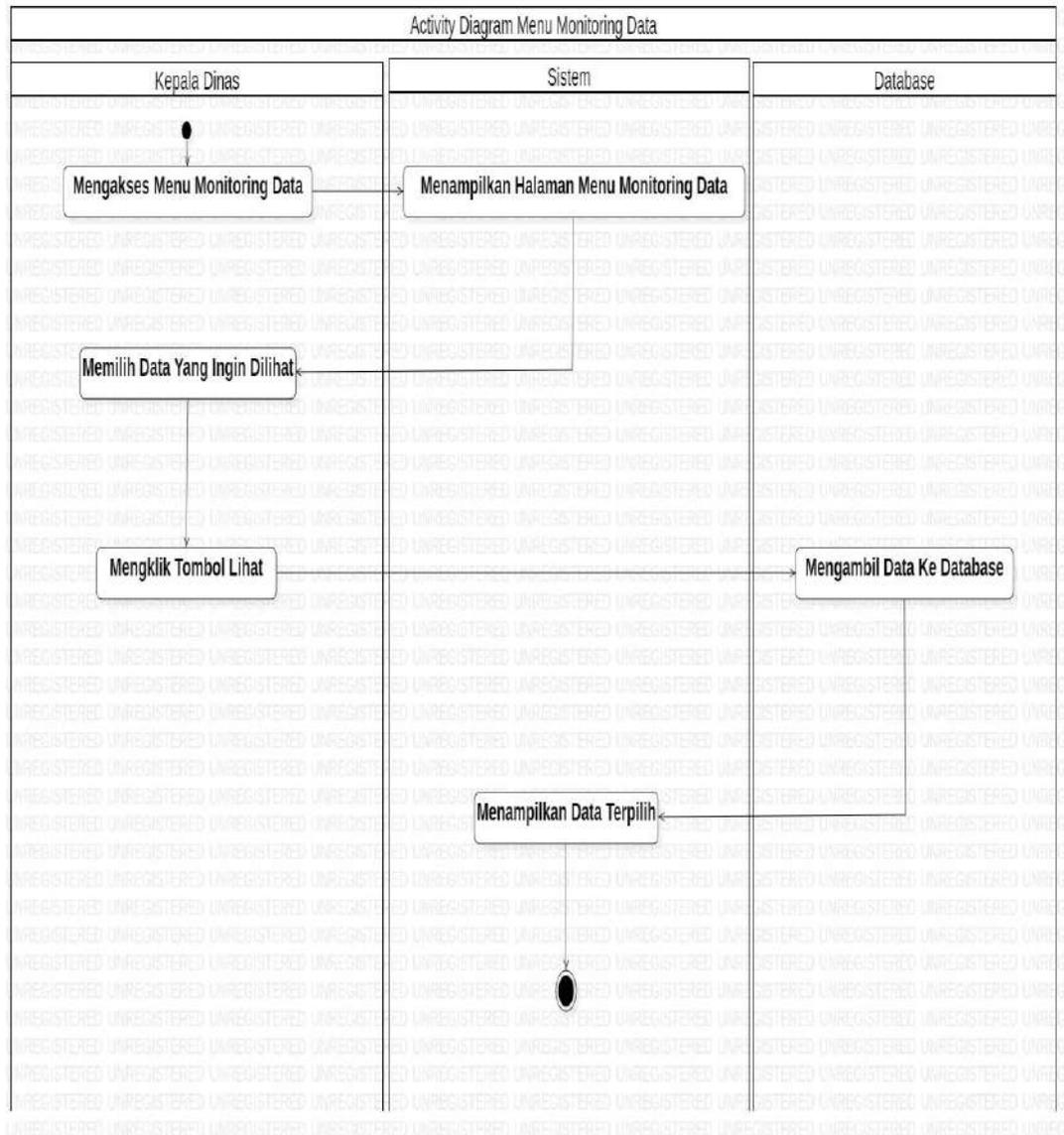


**Gambar 3.20** Activity Diagram Input Data Mutasi Pegawai

Dalam *activity diagram* ini aktor atau seorang pegawai dapat memilih *menu input data* mutasi pegawai dan selanjutnya sistem menampilkan halaman *input data* mutasi pegawai. Setelah sistem menampilkan halaman untuk *input* mutasi, maka tahapan berikutnya adalah pegawai mengisi *form* data mutasi yang sudah tersedia di halaman *input* mutasi pegawai, berikutnya sistem menyimpan data mutasi pegawai yang sudah dimasukkan oleh pegawai secara otomatis.

### 3.4.2.19 Activity Diagram Monitoring Data Kepala Dinas

Activity Diagram Monitoring Data Kepala Dinas disajikan pada Gambar 3.21

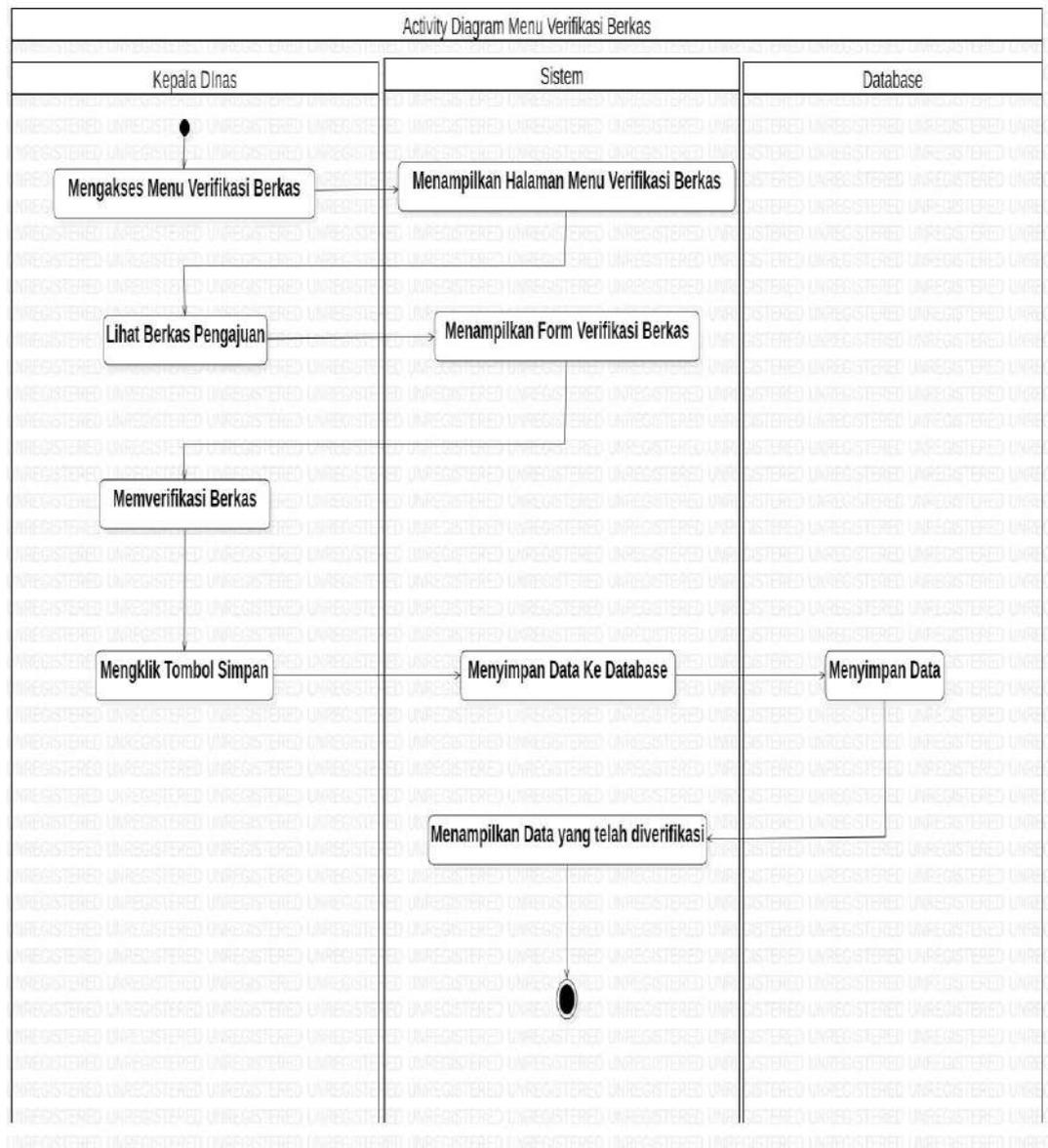


**Gambar 3.21** Activity Diagram Monitoring Data Kepala Dinas

Keapa Dinas dalam *activity diagram* diatas dapat mengakses halaman *monitoring* data, setelah itu sistem menampilkan halaman *monitoring* data. Selanjutnya kadis dapat melihat data pegawai secara lengkap dari masing-masing pegawai.

### 3.4.2.20 Activity Diagram Verifikasi Berkas

Activity Diagram Verifikasi Berkas disajikan pada Gambar 3.22



**Gambar 3.22** Activity Diagram Verifikasi Berkas

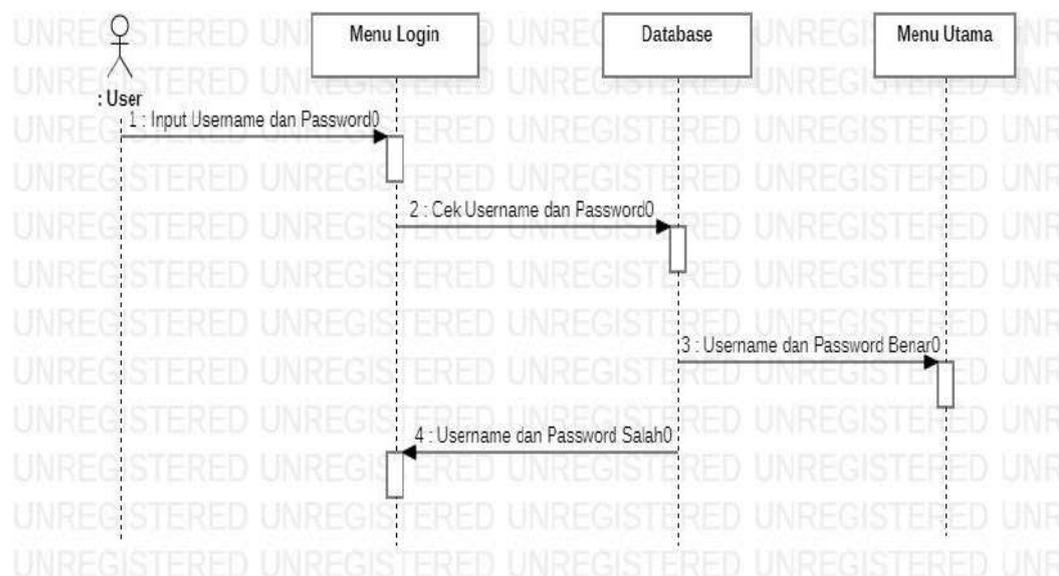
Dalam *activity diagram* diatas peran kepala dinas adalah memverifikasi berkas pengajuan dari masing-masing pegawai yaitu berupa berkas pengajuan cuti ataupun berkas pengajuan mutasi pegawai.

### 3.4.3 Sequence Diagram

*Sequence Diagram* menggambarkan interaksi antar objek didalam dan di sekitar sistem termasuk pengguna, serta *display* berupa *message* yang digambarkan terhadap waktu. *Sequence Diagram* dalam perancangan Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta adalah sebagai berikut :

#### 3.4.3.1 Sequence Diagram Login User

*Sequence Diagram Login User* dapat dilihat pada Gambar 3.23

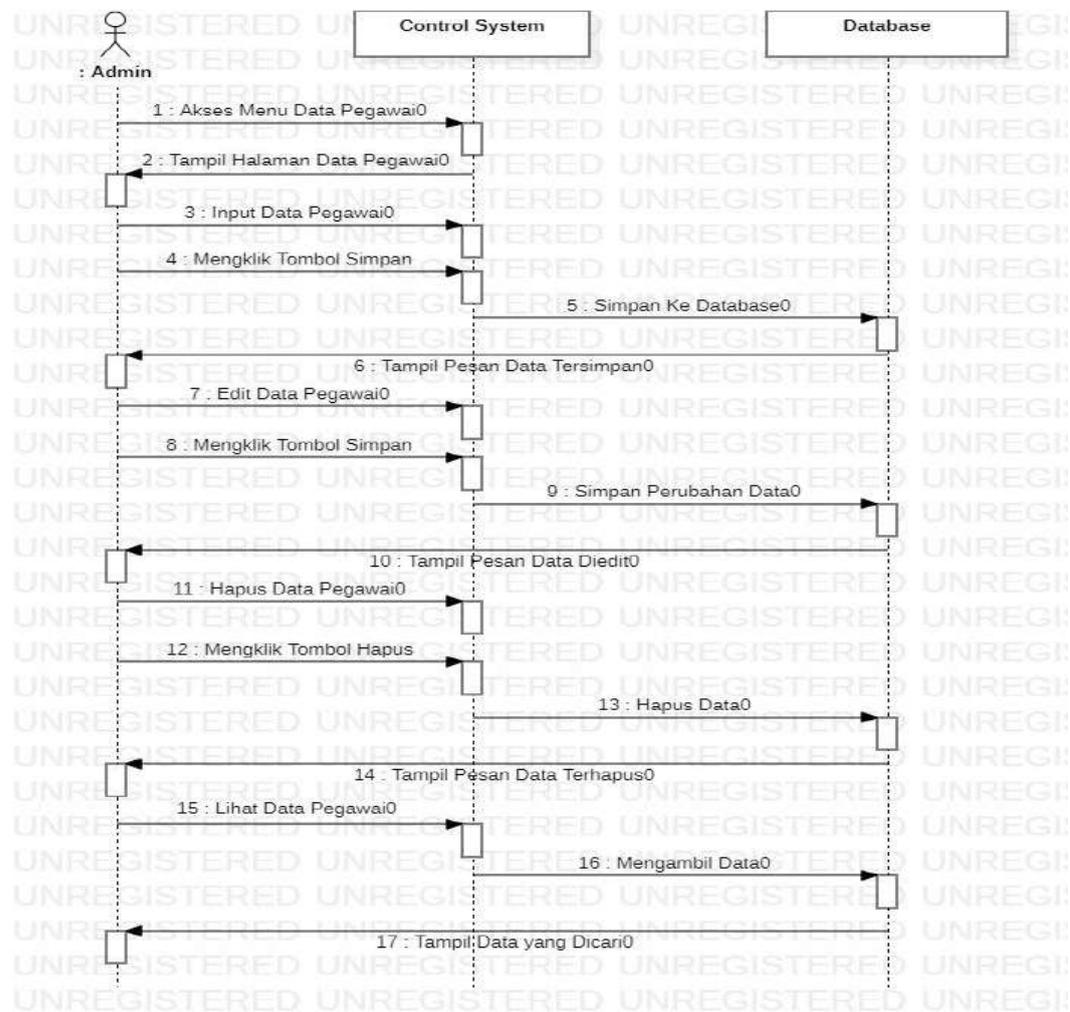


**Gambar 3.23** *Sequence Diagram Login User*

*User* dapat mengakses halaman *login*, berikutnya *user* memasukkan *username* dan *password*, setelah itu sistem mengecek *username* dan *password* yang dimasukkan oleh aktor sesuai dengan *database login* atau tidak. Jika *username* dan *password* benar maka sistem menampilkan *menu utama* dari sistem ini, namun jika *username* dan *password* salah maka sistem meminta *user* untuk memasukkan *username* dan *password* kembali.

### 3.4.3.2 Sequence Diagram Kelola Data Pegawai

Sequence Diagram Kelola Data Pegawai disajikan pada Gambar 3.24



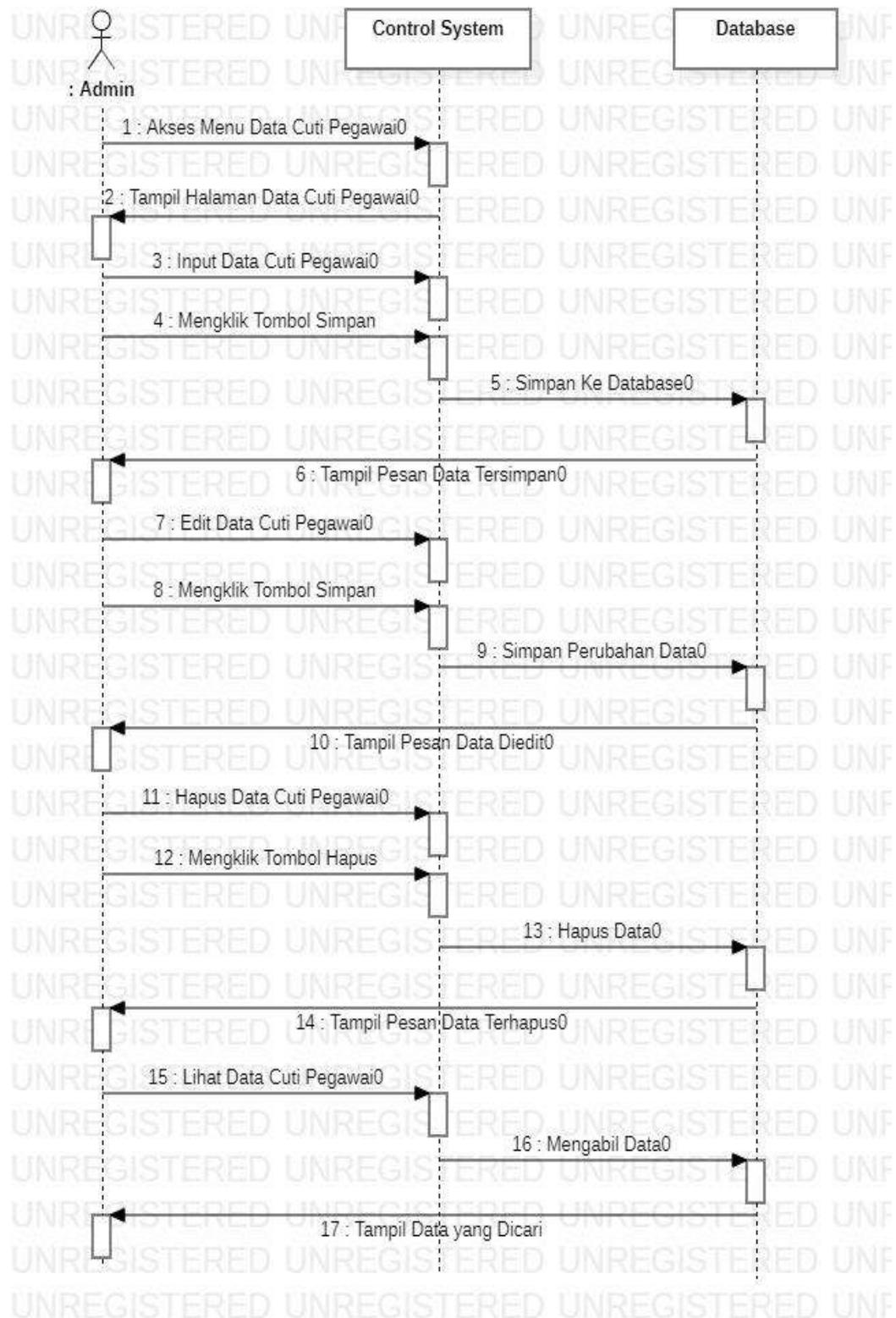
**Gambar 3.24** Sequence Diagram Kelola Data Pegawai

*Admin* dalam *sequence diagram* ini dapat mengakses halaman kelola data pegawai, setelah itu sistem menampilkan halaman data pegawai. Jika sistem telah menampilkan halaman untuk kelola data pegawai maka *admin* dapat menambah, mengedit, menghapus, mencari dan menyimpan data pegawai. Selanjutnya *admin* dapat mengklik *button* simpan untuk menyimpan data, maka data tersebut tersimpan ke dalam *database* serta sistem menampilkan pesan data tersebut berhasil disimpan.

Berikutnya jika *admin* mengklik *button edit* untuk memperbaharui data maka data yang terdapat didalam *database* menyimpan serta telah diperbaharui, kemudian sistem menampilkan pesan data tersebut berhasil diperbaharui, namun jika *admin* mengklik *button* hapus untuk menghapus data, maka data tersebut terhapus dari *database* dan sistem menampilkan pesan data tersebut berhasil dihapus. Bila *admin* memasukkan kata kunci untuk mencari data dan mengklik *button* cari atau menekan tombol cari maka sistem mencari data yang diperlukan dari *database* serta sistem menampilkan data yang dicari oleh *admin*.

### 3.4.3.3 Sequence Diagram Kelola Data Cuti Pegawai

Sequence Diagram Kelola Data Cuti Pegawai dapat diperhatikan Gambar 3.25



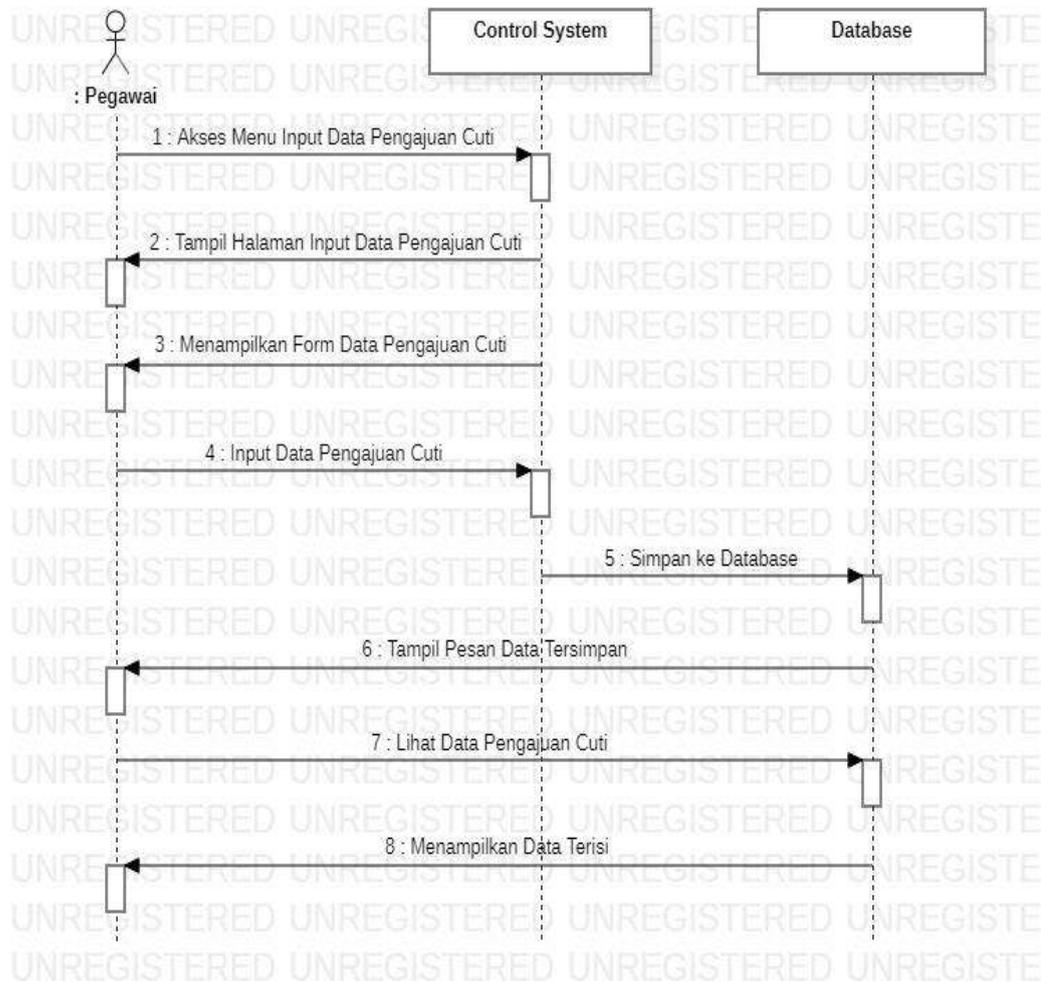
**Gambar 3.25** Sequence Diagram Kelola Data Cuti Pegawai

*Admin* dalam *sequence diagram* ini dapat mengakses halaman kelola data cuti pegawai, setelah itu sistem menampilkan halaman data cuti pegawai. Jika sistem telah menampilkan halaman untuk kelola data cuti pegawai maka *admin* dapat menambah, mengedit, menghapus, mencari dan menyimpan data pegawai. Selanjutnya *admin* dapat mengklik *button* simpan untuk menyimpan data, maka data tersebut tersimpan ke dalam *database* serta sistem menampilkan pesan data tersebut berhasil disimpan.

Berikutnya jika *admin* mengklik *button edit* untuk memperbaharui data maka data yang terdapat didalam *database* menyimpan serta telah diperbaharui, kemudian sistem menampilkan pesan data tersebut berhasil diperbaharui, namun jika *admin* mengklik *button* hapus untuk menghapus data, maka data tersebut terhapus dari *database* dan sistem menampilkan pesan data tersebut berhasil dihapus. Bila *admin* memasukkan kata kunci untuk mencari data dan mengklik *button* cari atau menekan tombol cari maka sistem mencari data yang diperlukan dari *database* serta sistem menampilkan data yang dicari oleh *admin*.

### 3.4.3.4 Sequence Diagram Input Data Pengajuan Cuti

Sequence Diagram Input Data Pengajuan Cuti disajikan pada Gambar 3.26

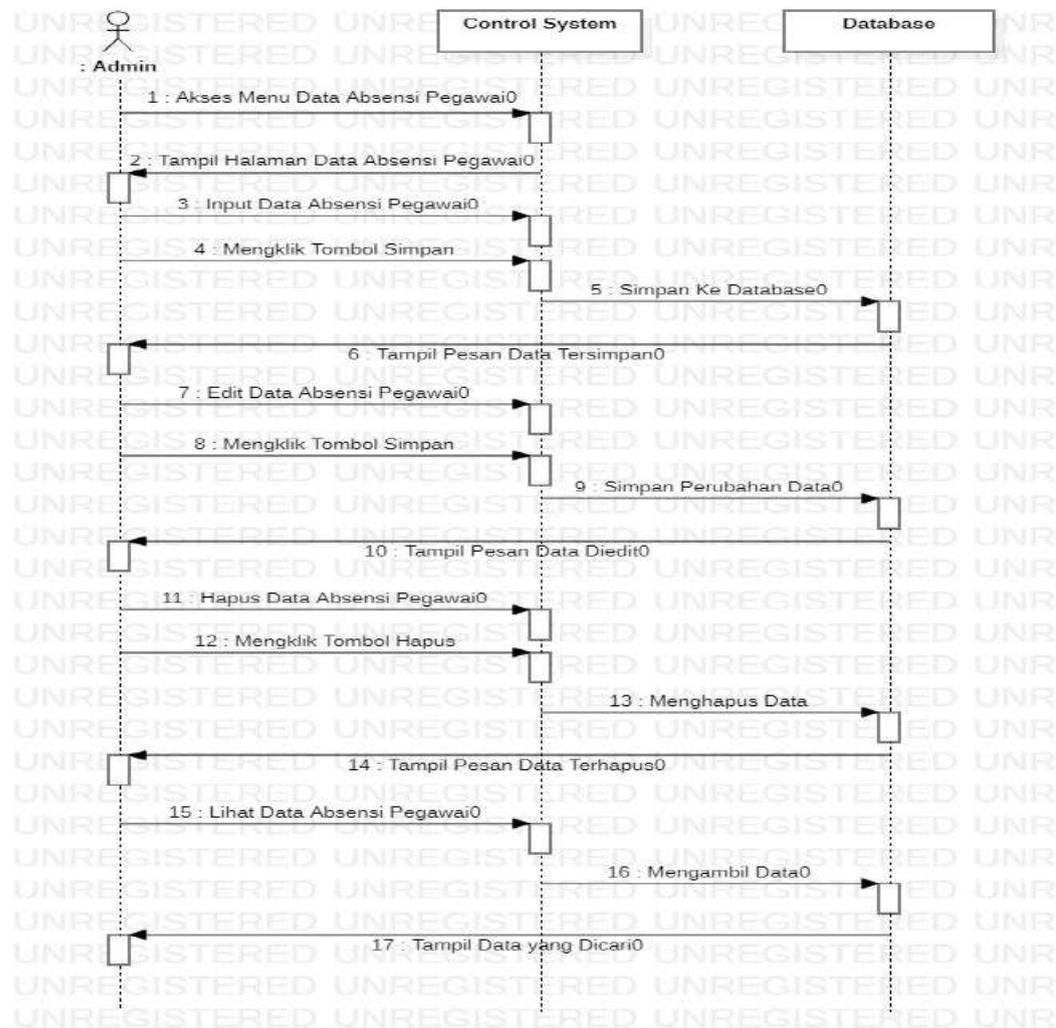


**Gambar 3.26** Sequence Diagram Input Data Pengajuan Cuti

Aktor pegawai dalam *sequence diagram* ini dapat memilih *menu input* data mutasi pegawai, setelah sistem menampilkan halaman *input* data mutasi pegawai. Berikutnya pegawai mengisi *form* data mutasi, jika *form* data mutasi telah dimasukkan data dengan baik, maka tahapan berikutnya yakni pegawai mengklik *button* simpan untuk menyimpan data agar data dapat tersimpan dengan baik didalam *database*. Jika data sudah tersimpan selanjutnya sistem menampilkan pesan data tersebut berhasil disimpan.

### 3.4.3.5 Sequence Diagram Kelola Data Absensi Pegawai

Sequence Diagram Kelola Data Absensi Pegawai disajikan pada Gambar 3.27



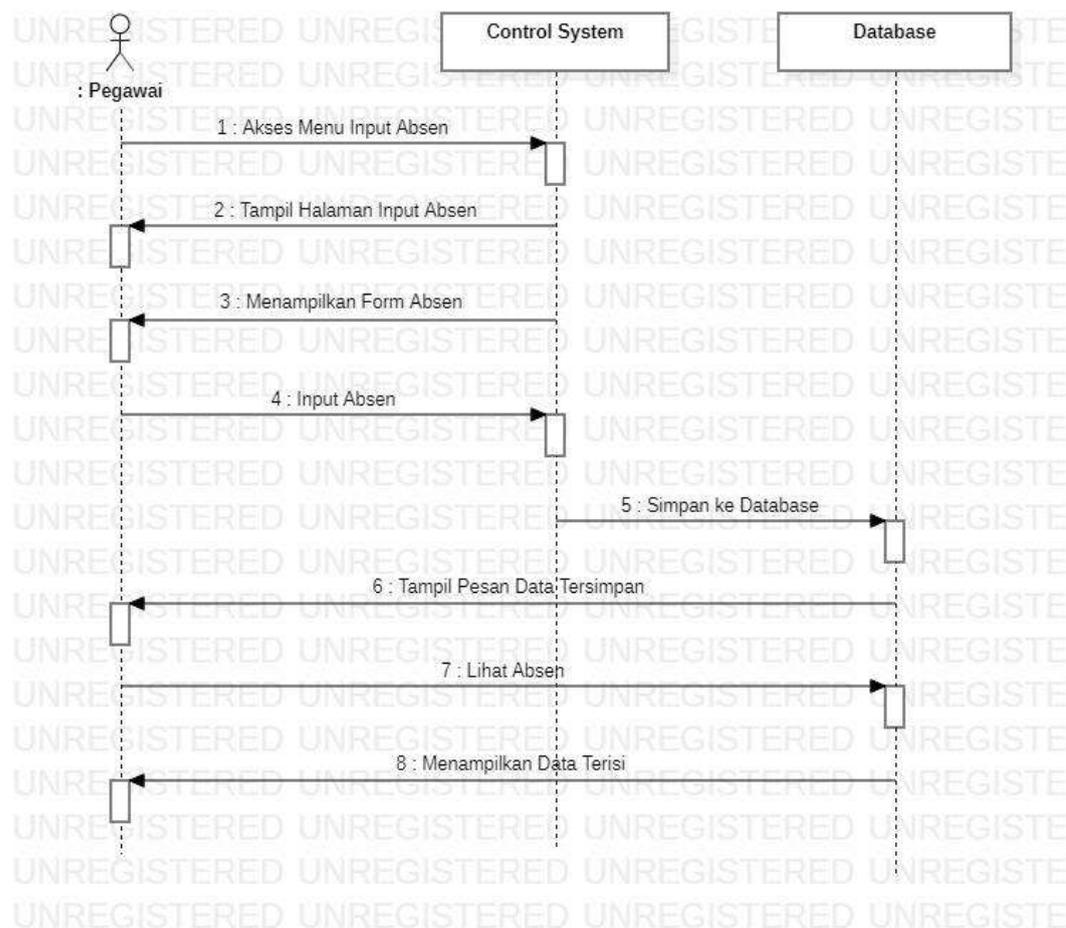
**Gambar 3.27** Sequence Diagram Kelola Data Absensi Pegawai

Aktor atau *admin* dalam *sequence diagram* ini dapat mengakses halaman kelola data absensi pegawai, setelah itu sistem menampilkan halaman data absensi pegawai. Jika sistem telah menampilkan halaman untuk kelola data absensi pegawai maka *admin* dapat menambah, mengedit, menghapus, mencari dan menyimpan data pegawai. Selanjutnya *admin* dapat mengklik *button* simpan untuk menyimpan data, maka data tersebut tersimpan ke dalam *database* serta sistem menampilkan pesan data tersebut berhasil disimpan. Berikutnya jika *admin*

mengklik *button edit* untuk memperbaharui data maka data yang terdapat didalam *database* menyimpan serta telah diperbaharui, kemudian sistem menampilkan pesan data tersebut berhasil diperbaharui, namun jika *admin* mengklik *button hapus* untuk menghapus data, maka data tersebut terhapus dari *database* dan sistem menampilkan pesan data tersebut berhasil dihapus. Bila *admin* memasukkan kata kunci untuk mencari data dan mengklik *button cari* atau menekan tombol cari maka sistem mencari data yang diperlukan dari *database* serta sistem menampilkan data yang dicari oleh *admin*.

### 3.4.3.6 Sequence Diagram Input Absen

Pada *Sequence Diagram Input Absen* dapat dilihat pada Gambar 3.28

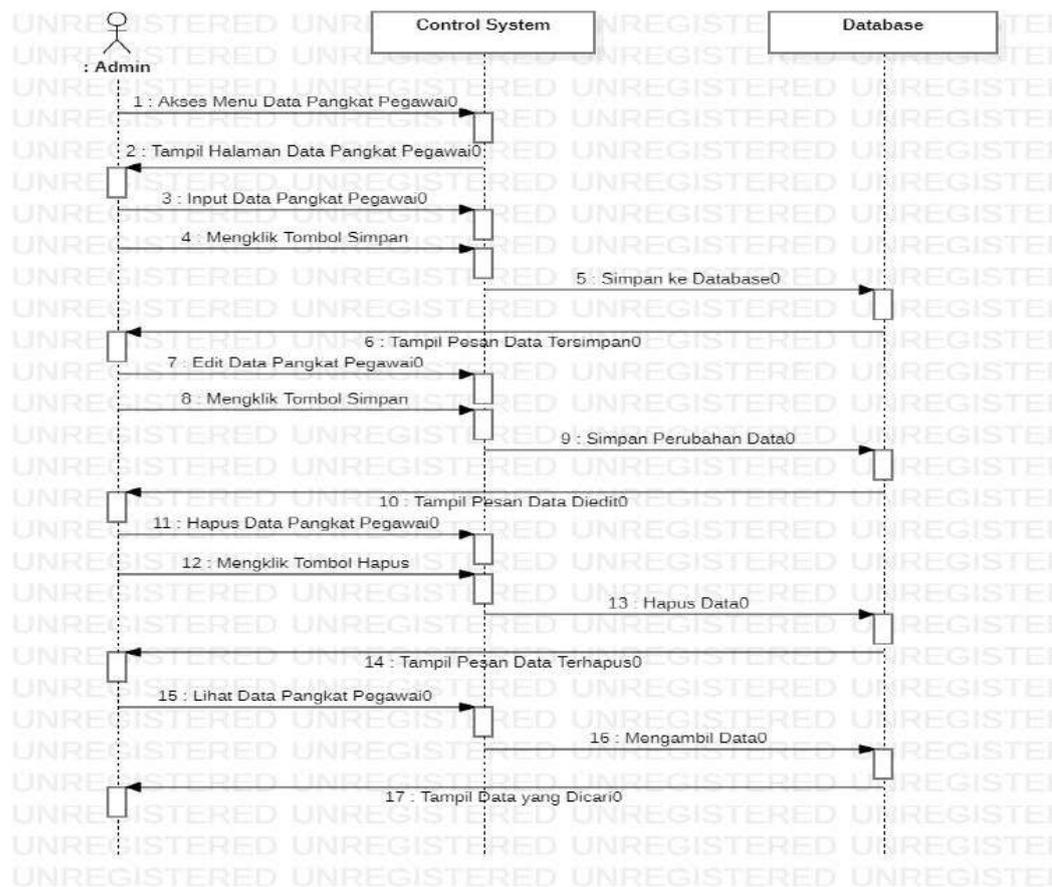


**Gambar 3.28** *Sequence Diagram Input Absen*

Dalam *sequence diagram* diatas aktor pegawai dapat memilih *menu input* absen untuk memasukkan absen sakit atau ijin, setelah pegawai mengklik *menu input* absen, maka selanjutnya sistem menampilkan halaman *input* absen. Setelah sistem menampilkan halaman *input* absen tahapan berikutnya pegawai dapat mengisi *form* absen, jika pegawai sudah mengisi form absen maka pegawai mengklik *button* simpan untuk menyimpan data agar data tersimpan dengan baik didalam *database* serta sistem menampilkan pesan bahwa data yang dimasukkan berhasil tersimpan.

### 3.4.3.7 Sequence Diagram Kelola Data Pangkat Pegawai

*Sequence Diagram* Kelola Data Pangkat Pegawai disajikan pada Gambar 3.29



**Gambar 3.29** *Sequence Diagram* Kelola Data Pangkat Pegawai

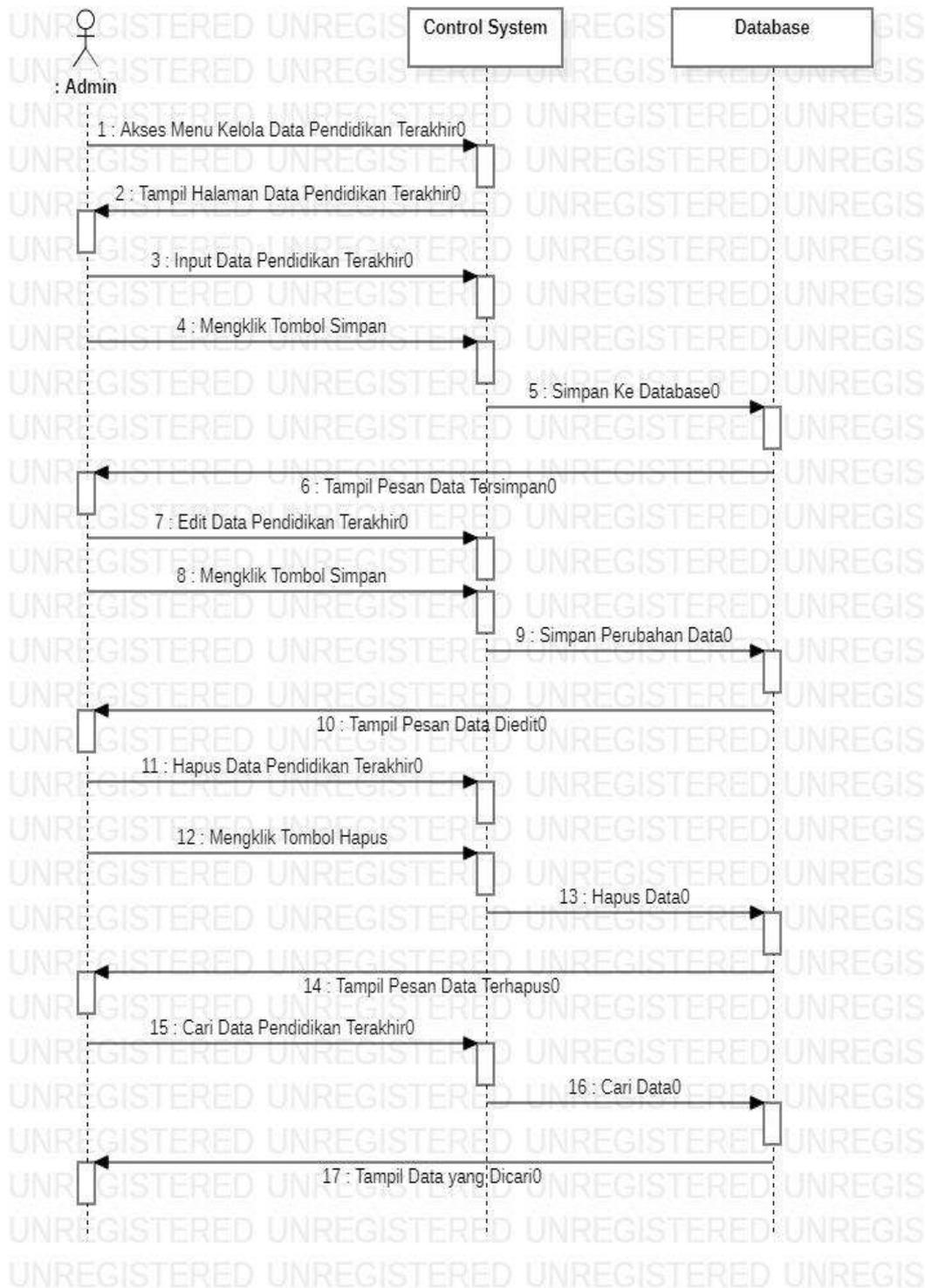
Seorang *admin* dalam *sequence diagram* ini dapat mengakses halaman kelola data pangkat pegawai, setelah itu sistem menampilkan halaman data pangkat pegawai. Jika sistem telah menampilkan halaman untuk kelola data pangkat pegawai maka *admin* dapat menambah, mengedit, menghapus, mencari dan menyimpan data pegawai. Selanjutnya *admin* dapat mengklik *button* simpan untuk menyimpan data, maka data tersebut tersimpan ke dalam *database* serta sistem menampilkan pesan data tersebut berhasil disimpan.

Berikutnya jika *admin* mengklik *button edit* untuk memperbaharui data maka data yang terdapat didalam *database* tersimpan serta telah diperbaharui, kemudian sistem menampilkan pesan bahwa data tersebut berhasil diperbaharui, namun jika *admin* mengklik *button* hapus untuk menghapus data, maka data tersebut terhapus dari *database* dan sistem menampilkan pesan data tersebut berhasil dihapus. Bila *admin* memasukkan kata kunci untuk mencari data dan mengklik *button* cari atau menekan tombol cari maka sistem mencari data yang diperlukan dari *database* serta sistem menampilkan data yang dicari oleh *admin*.

### 3.4.3.8 Sequence Diagram Kelola Data Pendidikan Terakhir

*Sequence Diagram* Kelola Data Pendidikan Terakhir dapat diperhatikan pada

Gambar 3.30

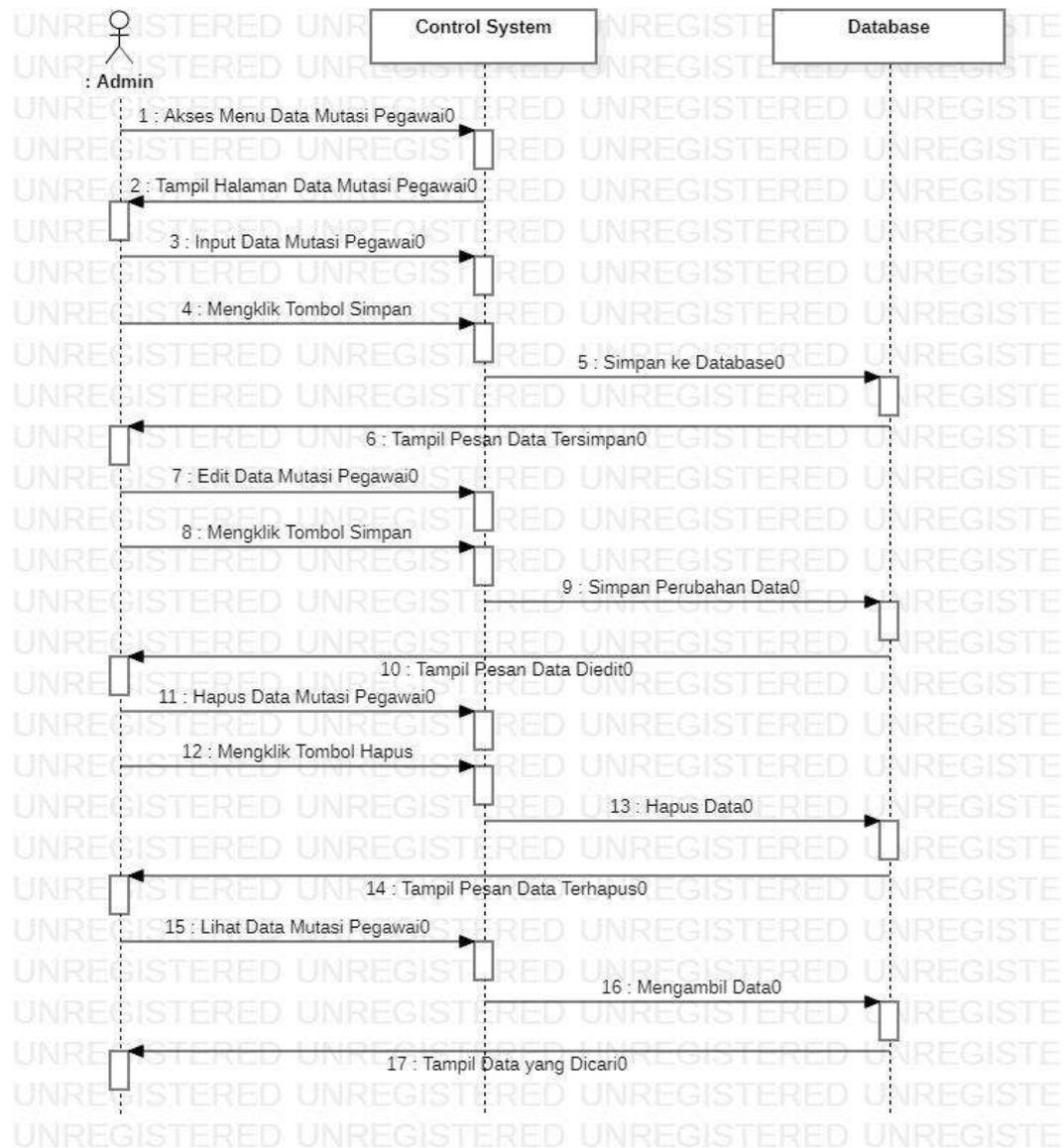


**Gambar 3.30** *Sequence Diagram* Kelola Data Pendidikan Terakhir

*Admin* dalam *sequence diagram* ini dapat mengakses halaman kelola data pendidikan terakhir, setelah itu sistem menampilkan halaman data pendidikan terakhir. Jika sistem telah menampilkan halaman untuk kelola data pendidikan terakhir maka *admin* dapat menambah, mengedit, menghapus, mencari dan menyimpan data pegawai. Selanjutnya *admin* dapat mengklik *button* simpan untuk menyimpan data, maka data tersebut tersimpan ke dalam *database* serta sistem menampilkan pesan data tersebut berhasil disimpan. Berikutnya jika *admin* mengklik *button edit* untuk memperbaharui data maka data yang terdapat didalam *database* menyimpan serta telah diperbaharui, kemudian sistem menampilkan pesan bahwa data tersebut berhasil diperbaharui, namun jika *admin* mengklik *button* hapus untuk menghapus data, maka data tersebut terhapus dari *database* dan sistem menampilkan pesan data tersebut berhasil dihapus. Bila *admin* memasukkan kata kunci untuk mencari data dan mengklik *button* cari atau menekan tombol cari maka sistem mencari data yang diperlukan dari *database* serta sistem menampilkan data yang dicari oleh *admin*.

### 3.4.3.9 Sequence Diagram Kelola Data Mutasi Pegawai

Sequence Diagram Kelola Data Mutasi Pegawai disajikan pada Gambar 3.31



**Gambar 3.31** Sequence Diagram Kelola Data Mutasi Pegawai

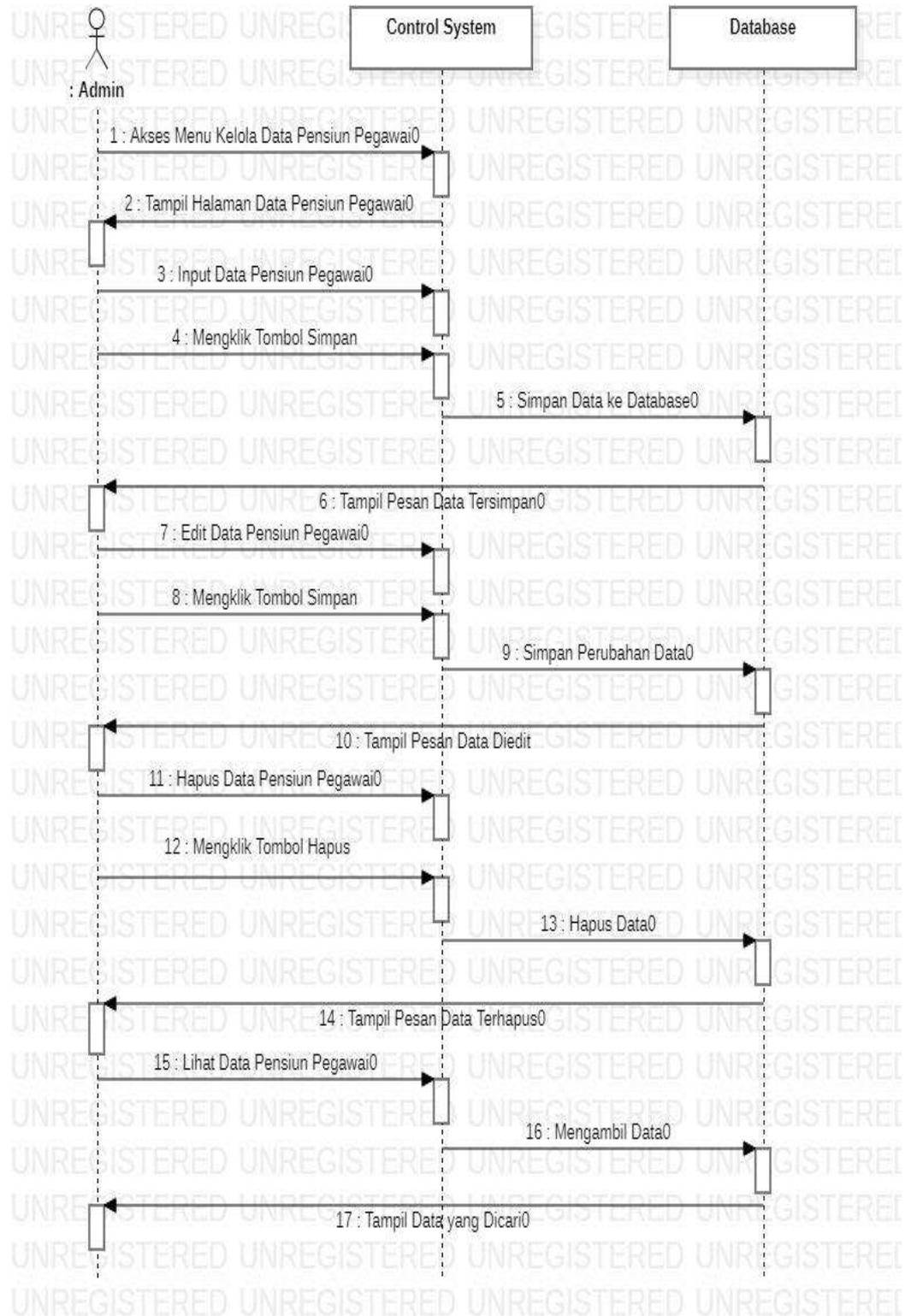
*Admin* dalam *sequence diagram* ini dapat mengakses halaman kelola data mutasi pegawai, setelah itu sistem menampilkan halaman data mutasi pegawai. Jika sistem telah menampilkan halaman untuk kelola data mutasi pegawai maka *admin* dapat menambah, mengedit, menghapus, mencari dan menyimpan data pegawai. Selanjutnya *admin* dapat mengklik *button* simpan untuk menyimpan data, maka

data tersebut tersimpan ke dalam *database* serta sistem menampilkan pesan data tersebut berhasil disimpan.

Berikutnya jika *admin* mengklik *button edit* untuk memperbaharui data maka data yang terdapat didalam *database* tersimpan serta telah diperbaharui, kemudian sistem menampilkan pesan data tersebut berhasil diperbaharui, namun jika *admin* mengklik *button* hapus untuk menghapus data, maka data tersebut terhapus dari *database* dan sistem menampilkan pesan data tersebut berhasil dihapus. Bila *admin* memasukkan kata kunci untuk mencari data dan mengklik *button* cari atau menekan tombol cari maka sistem mencari data yang diperlukan dari *database* serta sistem menampilkan data yang dicari oleh *admin*.

### 3.4.3.10 Sequence Diagram Kelola Data Pensiun Pegawai

Sequence Diagram Kelola Data Pensiun Pegawai disajikan pada Gambar 3.32



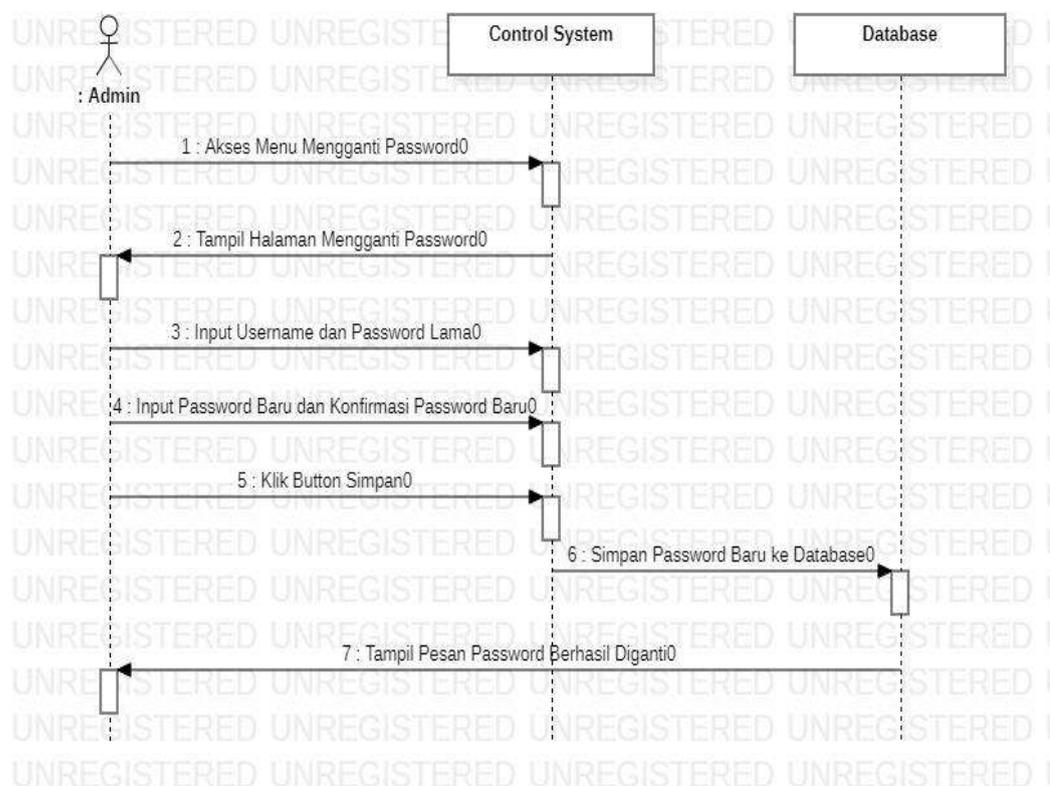
**Gambar 3.32** Sequence Diagram Kelola Data Pensiun Pegawai

Aktor *admin* dalam *sequence diagram* ini dapat mengakses halaman kelola data pensiun pegawai, setelah itu sistem menampilkan halaman data pensiun pegawai. Jika sistem telah menampilkan halaman untuk kelola data pensiun pegawai maka *admin* dapat menambah, mengedit, menghapus, mencari dan menyimpan data pegawai. Selanjutnya *admin* dapat mengklik *button* simpan untuk menyimpan data, maka data tersebut tersimpan ke dalam *database* serta sistem menampilkan pesan data tersebut berhasil disimpan.

Berikutnya jika *admin* mengklik *button edit* untuk memperbaharui data maka data yang terdapat didalam *database* menyimpan serta telah diperbaharui, kemudian sistem menampilkan pesan data tersebut berhasil diperbaharui, namun jika *admin* mengklik *button* hapus untuk menghapus data, maka data tersebut terhapus dari *database* dan sistem menampilkan pesan bahwa data tersebut berhasil dihapus. Bila *admin* memasukkan kata kunci untuk mencari data dan mengklik *button* cari atau menekan tombol cari maka sistem mencari data yang diperlukan dari *database* serta sistem menampilkan data yang dicari oleh *admin*.

### 3.4.3.11 Sequence Diagram Mengganti Password

Sequence Diagram Mengganti Password dapat diperhatikan pada Gambar 3.33

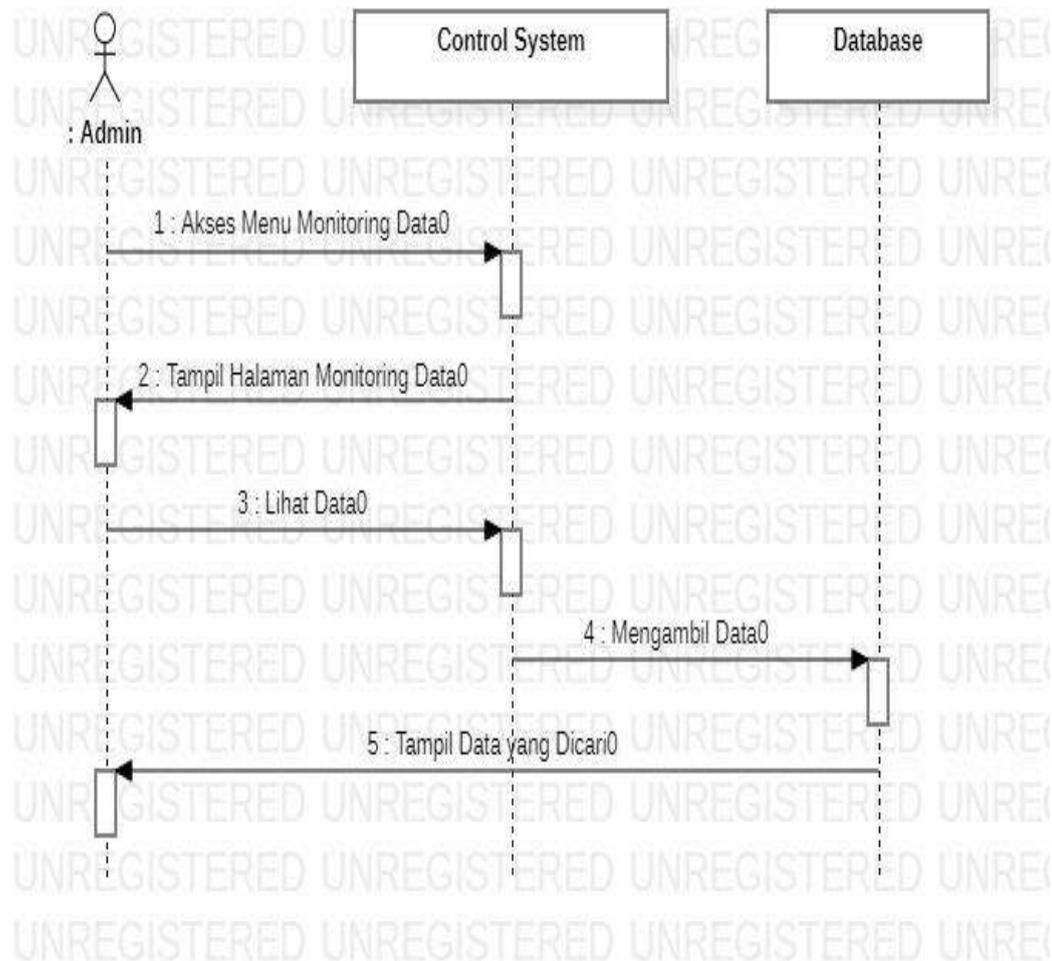


**Gambar 3.33** Sequence Diagram Mengganti Password

Aktor atau seorang *admin* dalam *sequence diagram* ini mengakses halaman mengganti *password*, selanjutnya sistem menampilkan halaman mengganti *password*. Setelah sistem menampilkan halaman untuk mengganti *password*, maka *admin* dapat memasukkan *username* dan *password* pegawai yang diganti dengan masukkan *password* lama, dan diperbaharui *password* baru serta konfirmasi *password* baru. Setelah itu *admin* mengklik *button* simpan agar *password* baru dapat tersimpan dengan baik. Kemudian sistem melakukan proses menyimpan *password* baru ke dalam *database*, setelah *proses* penyimpanan selesai, maka sistem menampilkan pesan ke *admin* bahwa *password* berhasil diperbaharui.

### 3.4.3.12 Sequence Diagram Monitoring Data

Sequence Diagram Monitoring Data dapat dilihat pada Gambar 3.34

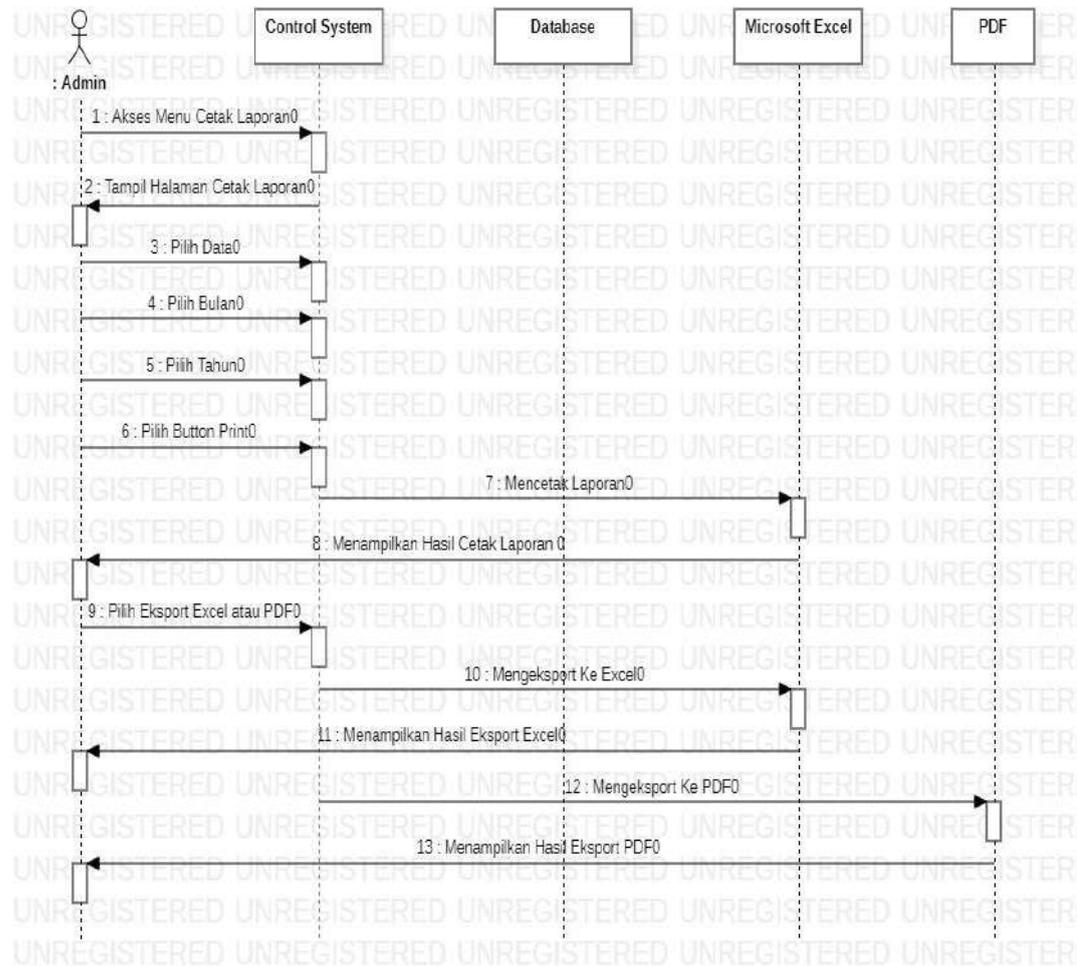


**Gambar 3.34** Sequence Diagram Monitoring Data

Aktor atau seorang *admin* dalam *sequence diagram* ini dapat mengakses halaman *monitoring data*, kemudian sistem menampilkan halaman *monitoring data* untuk melihat data yang masih belum dilengkapi oleh pegawai. Selanjutnya *admin* dapat melihat serta mencari data dengan mengklik nama *file data* dan mengecek data pegawai yang masih belum lengkap datanya serta dapat dilihat melalui diagram perkembangan pegawai, maka sistem menampilkan *file* yang diinginkan oleh *admin*, Selanjutnya *admin* memberikan pemberitahuan kepada pegawai untuk melengkapi datanya.

### 3.4.3.13 Sequence Diagram Cetak Laporan

Sequence Diagram Cetak Laporan dapat diperhatikan pada Gambar 3.35



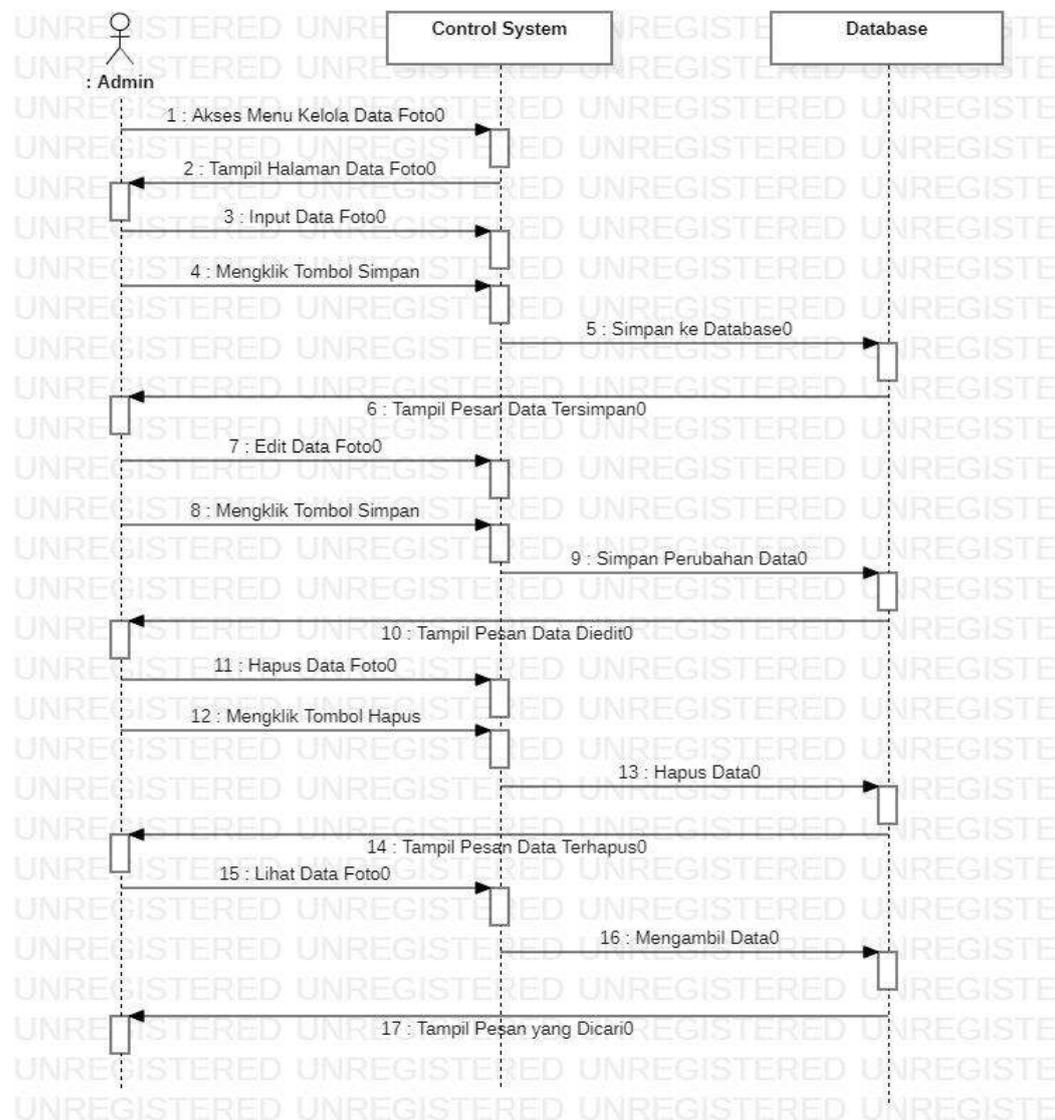
**Gambar 3.35** Sequence Diagram Cetak Laporan

Dalam *sequence diagram* ini seorang *admin* dapat mengakses halaman cetak laporan, setelah itu sistem menampilkan halaman cetak laporan. Jika sistem telah menampilkan halaman cetak laporan maka tahapan selanjutnya *admin* dapat memilih masing-masing data yang ingin dicetak, setelah itu *admin* juga dapat memilih data berdasarkan tahun, dan bulan yang ingin dicetak. Kemudian *admin*

memilih perintah *print* atau *ekspor* data dalam bentuk *pdf* atau *microsoft excel* untuk mencetak laporan. Jika *admin* mengklik tombol *print* maka sistem menampilkan hasil cetak dalam bentuk *print microsoft excel*, sedangkan jika *admin* mengklik tombol *ekspor* maka sistem memberikan pilihan dalam bentuk *format microsoft excel* atau *pdf* dan selanjutnya sistem menampilkan hasil laporan yang telah dipilih.

### 3.4.3.14 Sequence Diagram Kelola Data Foto

Sequence Diagram Kelola Data Foto dapat diperhatikan Gambar 3.36



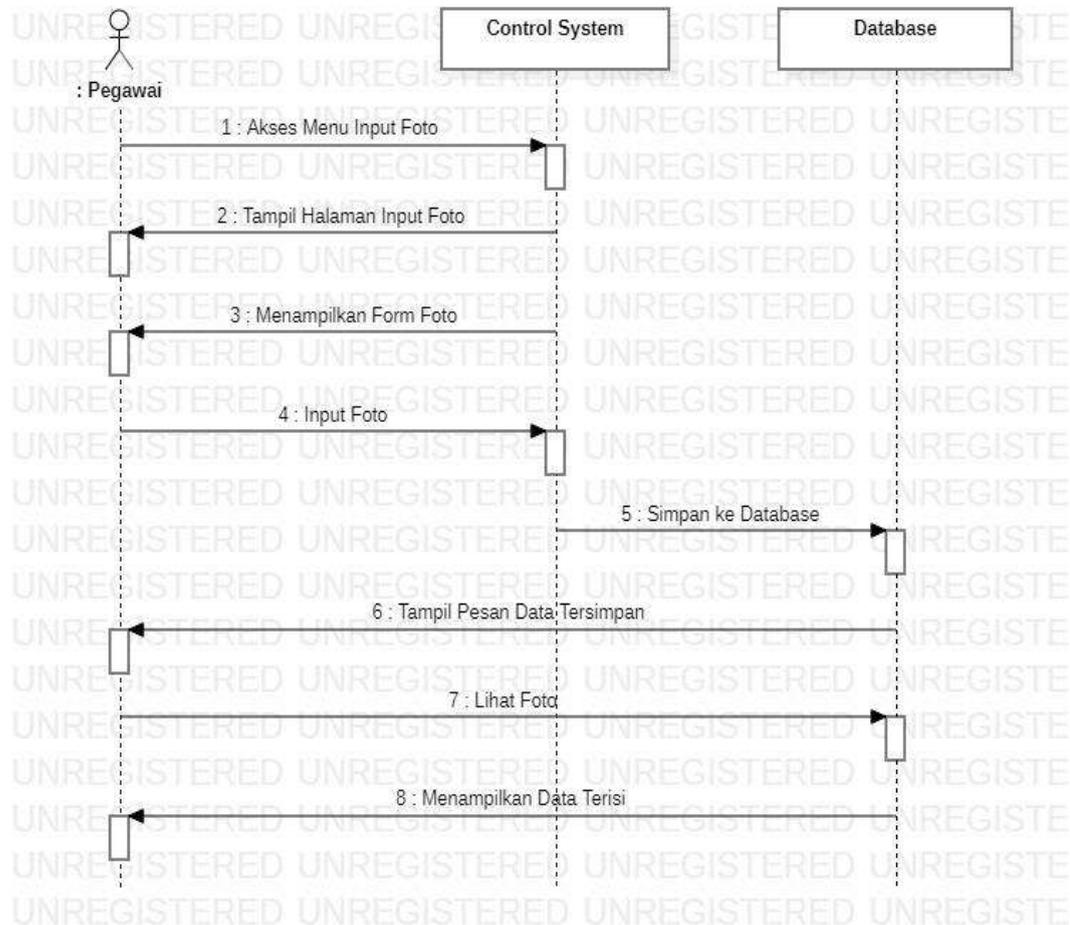
Gambar 3.36 Sequence Diagram Kelola Data Foto

*Admin* dalam *sequence diagram* ini dapat mengakses halaman kelola data foto, setelah itu sistem menampilkan halaman data foto. Jika sistem telah menampilkan halaman untuk kelola data foto maka *admin* dapat menambah, mengedit, menghapus, mencari dan menyimpan data foto pegawai. Selanjutnya *admin* dapat mengklik *button* simpan untuk menyimpan data, maka data tersebut tersimpan ke dalam *database* serta sistem menampilkan pesan data tersebut berhasil disimpan.

Berikutnya jika *admin* mengklik *button edit* untuk memperbaharui data maka data yang terdapat didalam *database* tersimpan serta telah diperbaharui, kemudian sistem menampilkan pesan data tersebut berhasil diperbaharui, namun jika *admin* mengklik *button* hapus untuk menghapus data, maka data tersebut terhapus dari *database* dan sistem menampilkan pesan data tersebut berhasil dihapus. Bila *admin* memasukkan kata kunci untuk mencari data dan mengklik *button* cari atau menekan tombol cari maka sistem mencari data yang diperlukan dari *database* serta sistem menampilkan data yang dicari oleh *admin*.

### 3.4.3.15 Sequence Diagram Input Foto

Sequence Diagram Input Foto disajikan pada Gambar 3.37

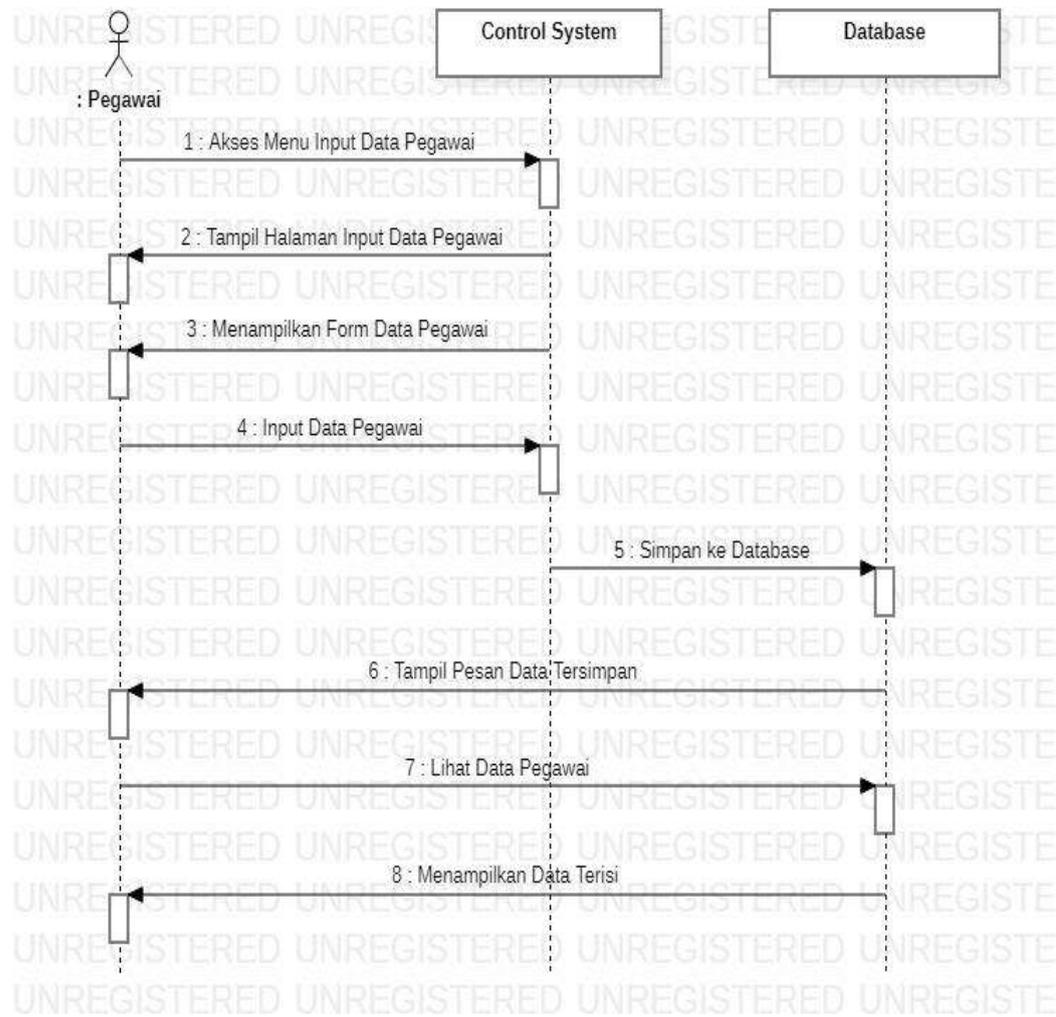


**Gambar 3.37** Sequence Diagram Input Foto

Dalam *sequence diagram* ini aktor pegawai dapat memilih *menu input foto*, maka sistem merespon dengan menampilkan halaman *input foto*. Setelah itu pegawai mengisi *form foto* yang sudah tersedia, setelah pegawai memasukkan *file foto* selanjutnya pegawai mengklik *button* simpan untuk menyimpan data foto yang telah dimasukkan tadi agar dapat tersimpan dengan baik, berikutnya sistem menampilkan pesan bahwa data berhasil disimpan.

### 3.4.3.16 Sequence Diagram Input Data Pegawai

Sequence Diagram Input Data Pegawai dapat diperhatikan pada Gambar 3.38

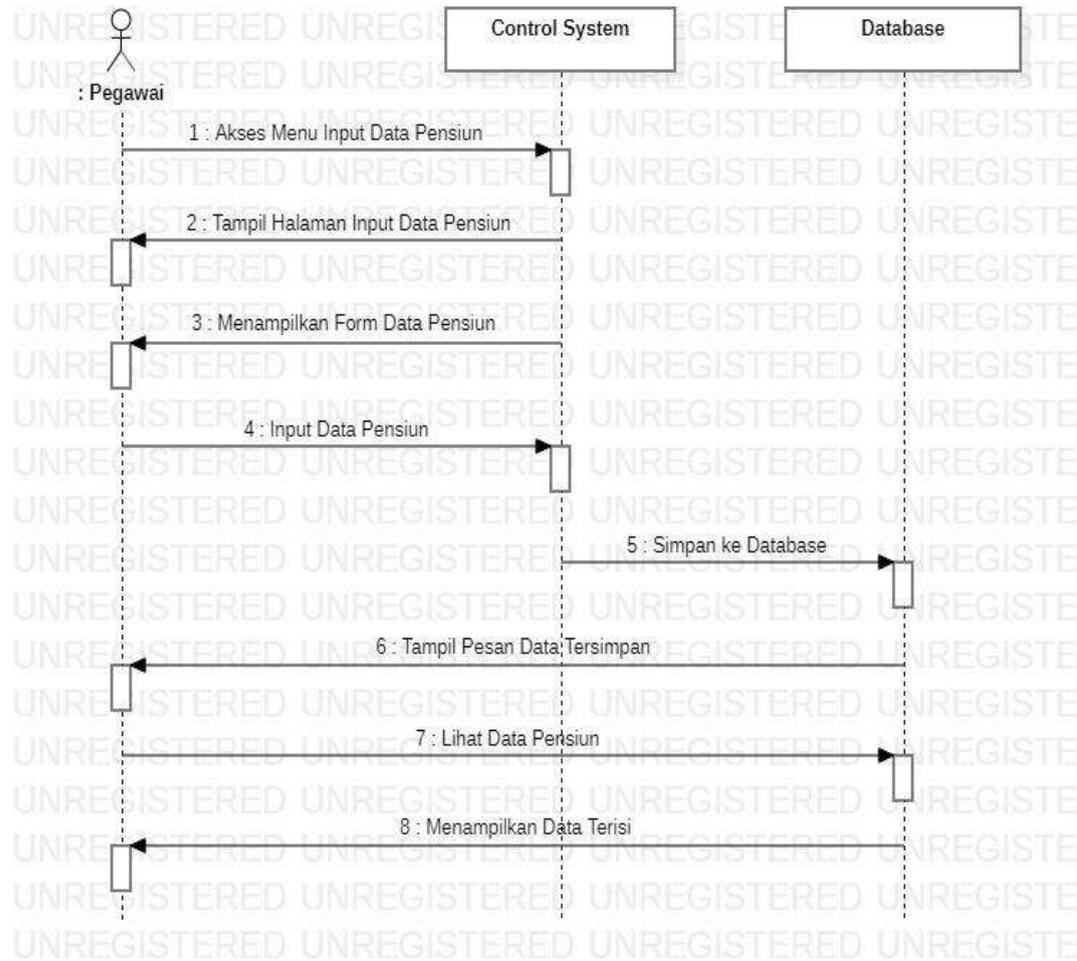


**Gambar 3.38** Sequence Diagram Input Data Pegawai

Aktor atau seorang pegawai dalam *sequence diagram* diatas dapat memilih *menu input* data pegawai, berikutnya sistem menampilkan halaman *input* data pegawai. Setelah halaman *input* data pegawai ditampilkan oleh sistem Kemudian pegawai dapat mengisi *form* data pegawai yang sudah tersedia, selanjutnya pegawai mengklik *button* simpan untuk menyimpan data agar data dapat tersimpan dengan baik, data yang telah dimasukkan tersimpan ke dalam *database* serta sistem menampilkan pesan data tersebut berhasil disimpan.

### 3.4.3.17 Sequence Diagram Input Data Pensiun

Sequence Diagram Input Data Pensiun dapat diperhatikan pada Gambar 3.39

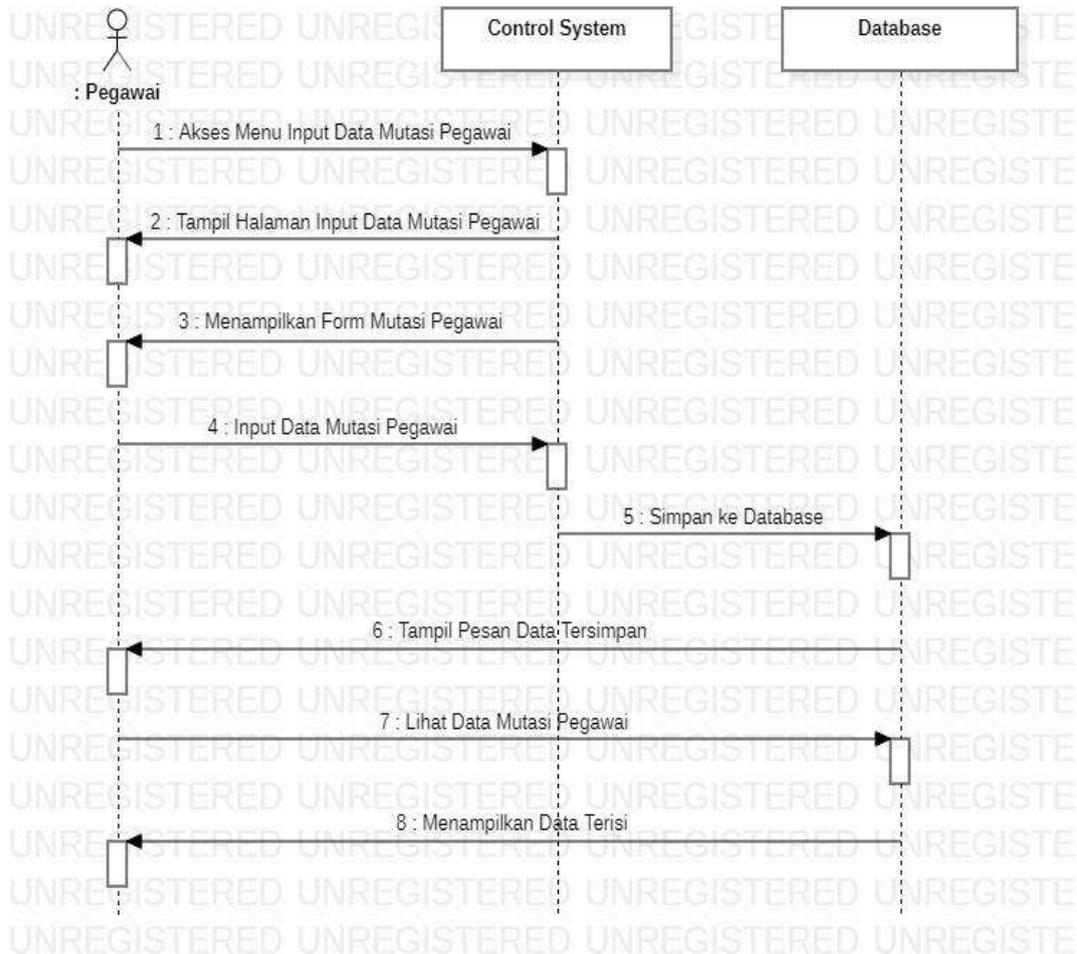


**Gambar 3.39** Sequence Diagram Input Data Pensiun

Dalam *sequence diagram* diatas seorang pegawai dapat memilih *menu input* data pensiun, selanjutnya sistem menampilkan halaman *input* data pensiun, dihalaman ini pegawai dapat mengisi data untuk masing-masing pegawai yang ingin mengajukan pensiun. Setelah itu pegawai mengisi *form* data pensiun yang sudah tersedia dalam sistem, setelah pegawai memasukkan data pensiun tahapan berikutnya adalah pegawai mengklik *button* simpan untuk menyimpan data agar data tersimpan dengan baik didalam *database*. Berikutnya sistem menampilkan pesan bahwa data tersebut berhasil tersimpan.

### 3.4.3.18 Sequence Diagram Input Data Mutasi Pegawai

Sequence Diagram Input Data Mutasi Pegawai disajikan pada Gambar 3.40

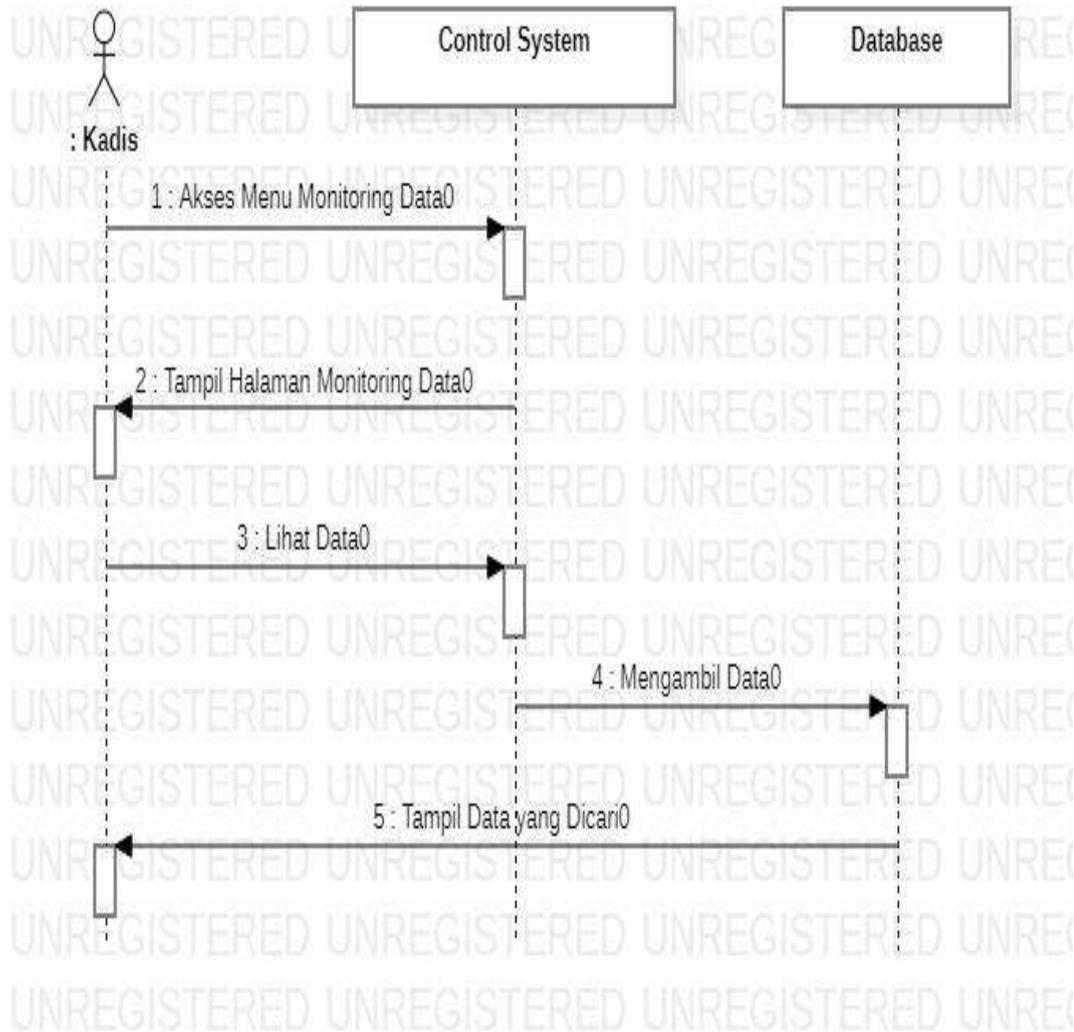


**Gambar 3.40** Sequence Diagram Input Data Mutasi Pegawai

Aktor pegawai dalam *sequence diagram* diatas dapat memilih *menu input data* mutasi pegawai, berikutnya sistem menampilkan halaman *input data* mutasi pegawai untuk memasukkan data bagi pegawai yang dimutasi. Setelah sistem menampilkan halaman *input data* mutasi pegawai, selanjutnya pegawai dapat mengisi *form data* mutasi pegawai, dan pegawai mengklik *button* simpan agar data dapat tersimpan dengan baik didalam *database*. Setelah pegawai memasukkan dan menyimpan data, tahapan berikutnya adalah sistem menampilkan pesan bahwa data tersebut berhasil tersimpan.

### 3.4.3.19 Sequence Diagram Monitoring Data Kadis

Sequence Diagram Monitoring Data Kadis dapat dilihat pada Gambar 3.41

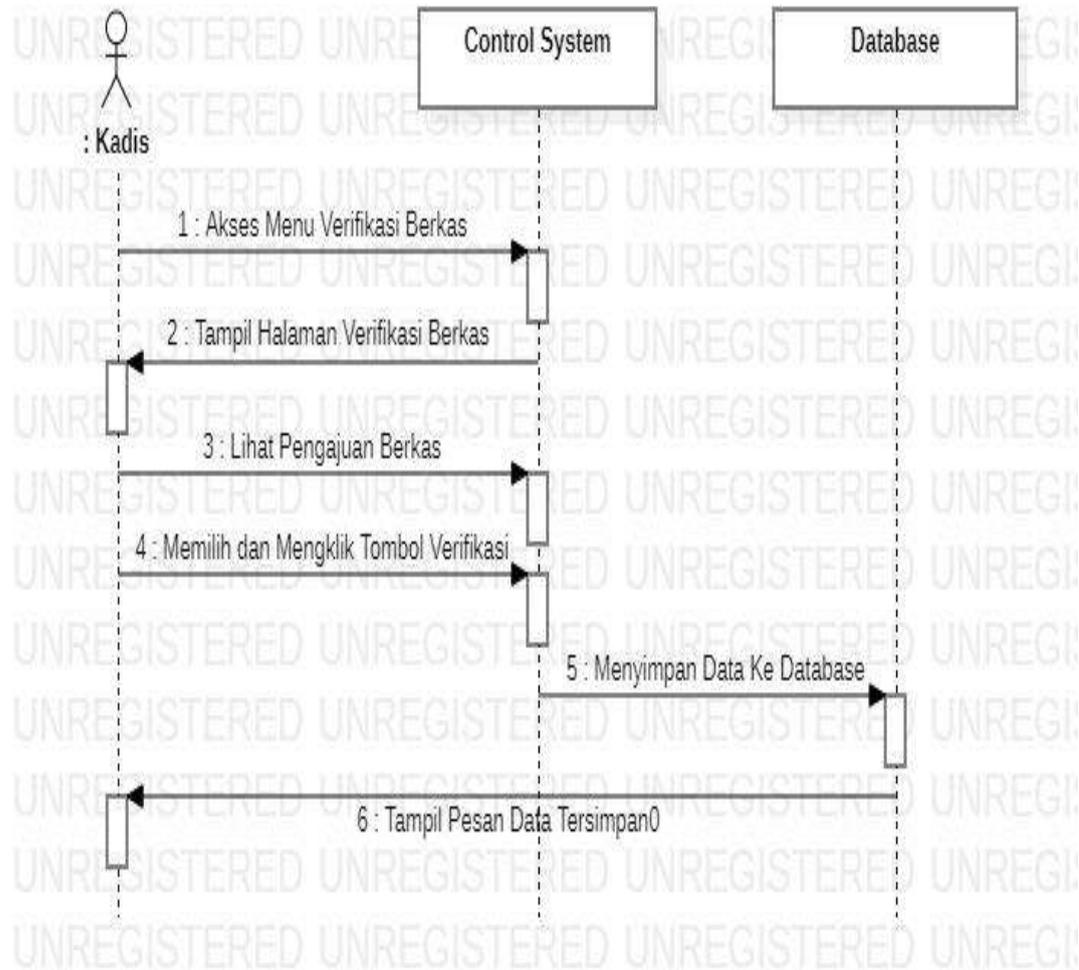


**Gambar 3.41** Sequence Diagram Monitoring Data Kadis

Aktor kadis dalam *sequence diagram* ini mengakses halaman *monitoring* data, kemudian sistem menampilkan halaman *monitoring* data fungsinya untuk melihat data keseluruhan pegawai. Selanjutnya kadis dapat melihat serta mencari data dengan mengklik nama *file* data salah satu pegawai serta dapat melihat perkembangan data melalui grafik perkembangan pegawai, sistem menampilkan *file* yang diinginkan oleh kadis.

### 3.4.3.20 Sequence Diagram Verifikasi Berkas

Sequence Diagram Verifikasi Berkas dapat dilihat pada Gambar 3.42

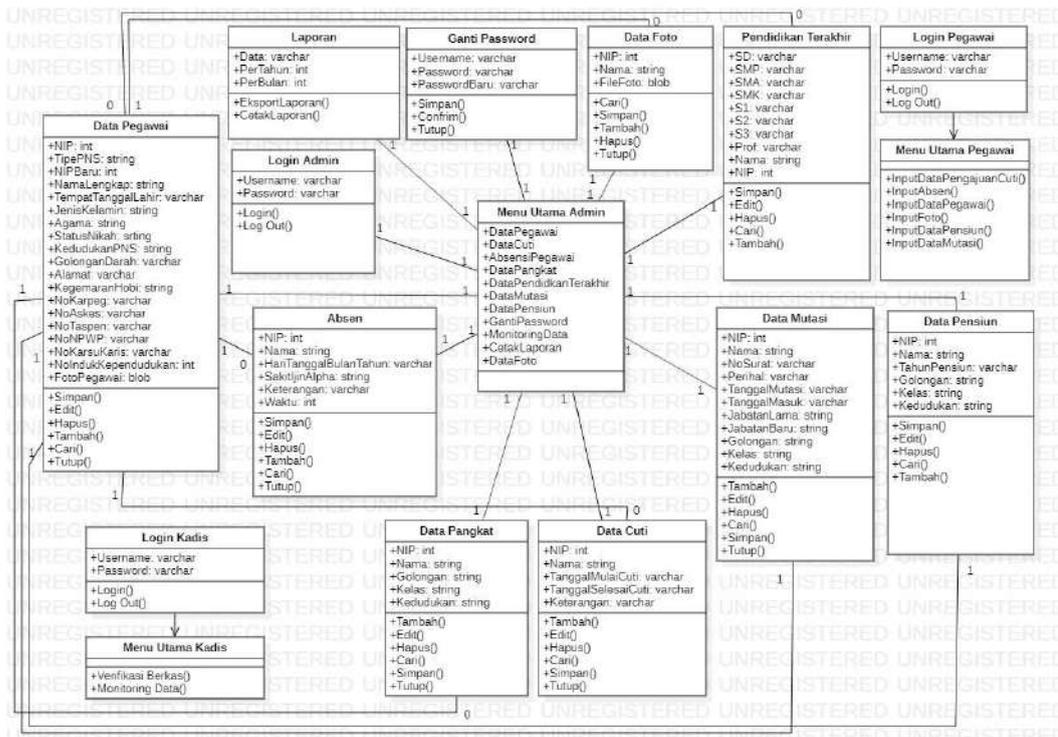


**Gambar 3.42** Sequence Diagram Verifikasi Berkas

Aktor kadis dalam *sequence diagram* ini mengakses halaman *verifikasi* berkas, kemudian sistem menampilkan halaman *verifikasi* berkas, selanjutnya kadis dapat memilih berkas pengajuan yang diajukan oleh pegawai untuk diverifikasi berkas pengajuannya yang diantaranya terdapat berkas pengajuan cuti, dan pengajuan mutasi. Setelah kadis memilih dan memverifikasi berkas pengajuan pegawai tersebut data berkas pengajuan tersebut tersimpan kedalam database secara otomatis sistem menampilkan pesan bahwa data berhasil tersimpan.

### 3.4.4 Class Diagram

*Class Diagram* perancangan Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta disajikan pada Gambar 3.43



**Gambar 3.43** *Class Diagram* Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta

*Class Diagram* digunakan untuk menggambarkan kumpulan dari *class-class* yang saling berhubungan. *Diagram* ini merupakan salah satu *diagram* yang paling umum ditemukan dalam pemodelan sistem berorientasi *object*. *Class* menggambarkan suatu keadaan sistem, sekaligus layanan untuk memanipulasi keadaan metode atau fungsi sehingga *class* memiliki tiga area pokok, yaitu : nama, atribut, dan metode. Selain itu setiap *class* yang ada dapat menjadi sebuah *form* saat pembuatan program salah satunya adalah *class diagram* yang digunakan pada perancangan Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta.

Sistem Informasi Kepegawian DISPORA memiliki 10 tabel, yaitu:

1. Tabel Data Pegawai

Tabel 3.26 Kamus data tabel pegawai

<i>Column</i>	<i>Type</i>
NIP	Int(18)
TipePNS	Varchar(10)
NipBaru	Int(18)
NamaLengkap	Varchar(30)
TempatTanggalLahir	String(11)
JenisKelamin	Varchar (2)
Agama	Varchar (6)
StatusNikah	Varchar (4)
KedudukanPNS	String(10)
GolonganDarah	Varchar(4)
Alamat	String(100)
Kegemaran/Hobi	Varchar(10)
NoKarpeg	Int(10)
NoAskes	Int(13)
NoTaspen	Int(10)
NoNPWP	Int(20)
NoKarsu/Karis	Int(11)
NoIndukKependudukan	Int(16)
FotoPegawai	Blob(1)

Tabel angkatan memiliki sembilanbelas kolom yaitu nip sebagai kunci dari tabel dan data pegawai. Tabel ini digunakan untuk menyimpan data pegawai dispora. Tabel data pegawai digunakan untuk menyimpan data keseluruhan pegawai dan memiliki operasi *create*, *read*, *edit* dan *delete*. Tabel data pegawai memiliki relasi dengan table-table yang terdapat dalam sistem kepegawian ini.

## 2. Tabel Laporan

Tabel 3.27 Kamus tabel laporan

<i>Column</i>	<i>Type</i>
Data	Varchar(100)
PerTahun	Int(11)
PerBulan	Int(11)

Tabel laporan memiliki tiga kolom yaitu data sebagai kunci dari tabel laporan untuk memilih jenis laporan yang ingin dicetak, data berisi seluruh data pegawai, pertahun dan perbulan yang merujuk pada data pegawai tabel data. Tabel laporan digunakan untuk mencetak data pegawai dan memiliki operasi *print* dan *export*. Tabel laporan memiliki relasi dengan tabel data pegawai, absen, foto, pendidikan terakhir, data mutasi, data pensiun, data pangkat, dan data cuti.

## 3. Tabel Ganti Password

Tabel 3.28 Kamus tabel ganti password

<i>Column</i>	<i>Type</i>
Username	Varchar(10)
Password	Varchar(10)
PasswordBaru	Varchar(10)

Tabel ganti *password* memiliki tiga kolom yaitu *username* sebagai kata kunci yang digunakan untuk *login*, *password* yang digunakan untuk membuka kunci sedangkan *password* baru untuk mengganti *password* lama. Tabel ganti *password* memiliki operasi *save* dan *close*.

#### 4. Tabel Data Foto

Tabel 3.29 Kamus data tabel foto

<i>Column</i>	<i>Type</i>
NIP	Int(18)
Nama	Varchar(30)
FileFoto	Blob(1)

Tabel data foto memiliki tiga kolom yaitu nip kunci dari data foto, nama dan *file* foto berisi foto yang *diinputkan*. Tabel data foto digunakan untuk menyimpan data foto dan memiliki operasi *create*, *read*, *update* dan *delete*. Tabel data foto memiliki relasi dengan tabel data pegawai.

#### 5. Tabel Pendidikan Terakhir

Tabel 3.30 Kamus data tabel pendidikan terakhir

<i>Column</i>	<i>Type</i>
NIP	Int(18)
Nama	Varchar(30)
SD	Varchar(50)
SMP	Varchar(50)

Tabel 3.30 Kamus data tabel pendidikan terakhir (Lanjutan)

<i>Column</i>	<i>Type</i>
SMP	Varchar(50)
SMA/SMK	Varchar(50)
S1	Varchar(50)
S2	Varchar(50)
S3	Varchar(50)

Tabel pendidikan terakhir memiliki sembilan kolom yaitu nip sebagai kunci dari tabel pendidikan terakhir, nama berisi nama pegawai, sd sampai S3 berisi data pendidikan. Tabel pendidikan terakhir digunakan untuk menyimpan data Pendidikan pegawai dan memiliki operasi *create*, *read*, *update* dan *delete*. Tabel pendidikan terakhir memiliki relasi dengan tabel data pegawai.

## 6. Tabel Absen

Tabel 3.31 Kamus data tabel absen

<i>Column</i>	<i>Type</i>
NIP	Int(18)
Nama	Varchar(30)
HariTanggalBulanTahun	String(11)
SakitIjinAlpha	Varchar(4)
Keterangan	Varchar(11)
Waktu	Date(4)

Tabel absen memiliki enam kolom yaitu nip sebagai kunci dari tabel absen, nama berisi nama pegawai, haritanggalbulantahun berisi waktu, sakitijinalpha berisi keterangan, keterangan yang berisi keterangan lengkap, dan waktu merujuk pada pegisian data absen. Tabel absen digunakan untuk menyimpan data absensi pegawai dan memiliki operasi *create*, *read*, *update* dan *delete*. Tabel absen memiliki relasi dengan tabel data pegawai.

#### 7. Tabel Data Mutasi

Tabel 3.32 Kamus data tabel mutasi

<i>Column</i>	<i>Type</i>
NIP	Int(18)
Nama	Varchar(30)
NoSurat	String(10)
Perihal	Varchar(10)
TanggalMutasi	Date(10)
TanggalMasuk	Date(10)
JabatanLama	Varchar(10)
JabatanBaru	Varchar(10)
Golongan	Varchar(4)
Kelas	Varchar(6)
Kedudukan	Varchar(10)

Tabel mutasi memiliki sebelas kolom yaitu nip sebagai kunci dari tabel mutasi, nama berisi nama pegawai, nosurat yang berisi penomoran surat, perihal berisi

jenis kegiatan, tanggal mutasi berisi tanggal pemindahan pegawai, tanggal masuk keterangan mulai kerja, jabatan lama berisi penempatan posisi lama, jabatan baru berisi tentang penempatan posisi terbaru, golongan berisi huruf romawi, kelas berisi huruf abjad sebagai pembeda tingkatan kedudukan dan kedudukan berisi posisi penempatan. Tabel mutasi digunakan untuk menyimpan data mutasi pegawai dan memiliki operasi *create*, *read*, *update* dan *delete*. Tabel mutasi memiliki relasi dengan tabel data pegawai.

## 8. Tabel Data Pensiun

Tabel 3.33 Kamus data tabel pensiun

<i>Column</i>	<i>Type</i>
NIP	Int(18)
Nama	Varchar(30)
TahunPensiun	Date(10)
Golongan	Varchar(4)
Kelas	Varchar(6)
Kedudukan	Varchar(10)

Tabel pensiun memiliki enam kolom yaitu nip sebagai kunci dari tabel pensiun. Tabel pensiun digunakan untuk menyimpan data pensiun pegawai dan memiliki operasi *create*, *read*, *update* dan *delete*. Tabel pensiun memiliki relasi dengan tabel data pegawai.

## 9. Tabel Data Pangkat

Tabel 3.34 Kamus data tabel pangkat

<i>Column</i>	<i>Type</i>
NIP	Int(18)
Nama	Varchar(30)
Golongan	Varchar(4)
Kelas	Varchar(6)
Kedudukan	Varchar(10)

Tabel pangkat memiliki lima kolom yaitu nip sebagai kunci dari tabel dan merujuk pada tabel data pegawai. Tabel pangkat digunakan untuk menyimpan data pangkat pegawai dan memiliki operasi *create*, *read*, *update* dan *delete*. Tabel pangkat memiliki relasi dengan tabel data pegawai.

## 10. Tabel Data Cuti

Tabel 3.35 Kamus data tabel cuti

<i>Column</i>	<i>Type</i>
NIP	Int(18)
Nama	Varchar(30)
TanggalMulaiCuti	Date(10)
TanggalSelesaiCuti	Date(10)
Keterangan	Varchar(50)

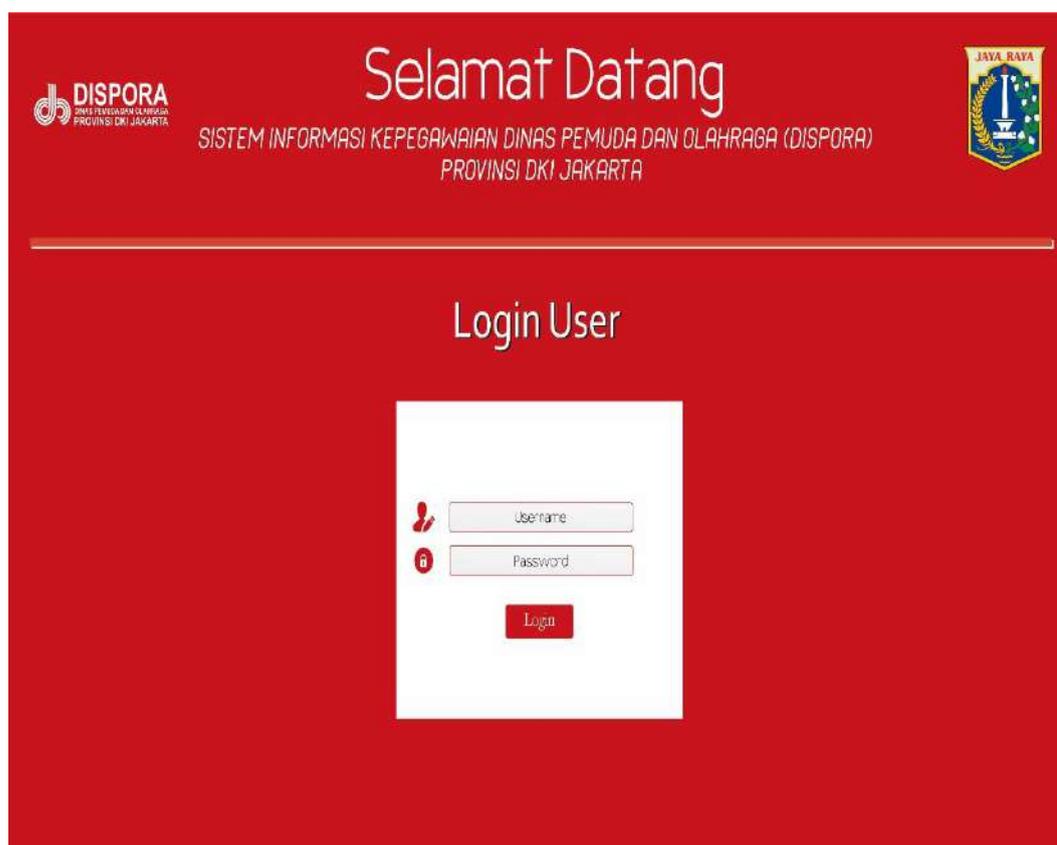
Tabel cuti memiliki enam kolom yaitu nip sebagai kunci dari tabel dan merujuk pada tabel data pegawai, tanggalmulaicuti yang berisi waktu pengabilan mulai cuti sedangkan tanggalselesaicuti yang berisi batas berakhirnya masa cuti. Tabel cuti digunakan untuk menyimpan data cuti pegawai dan memiliki operasi *create*, *read*, *update* dan *delete*. Tabel cuti memiliki relasi dengan tabel data pegawai.

#### **3.4.5 Rancangan *Input* untuk Akses *Admin***

Rancangan *input* untuk akses *admin* digunakan oleh *admin* untuk mengelola data pegawai, mengelola data cuti pegawai, mengelola data absensi pegawai, mengelola data pangkat pegawai, mengelola data pendidikan terakhir, mengelola data mutasi pegawai, mengelola data pensiun pegawai, mengganti *password*, *me-monitoring* data yang telah dimasukkan, mencetak laporan, mengelola data foto. Berikut desain *interface* pada rancangan *input* untuk akses *admin* :

### 3.4.5.1 Desain *Interface Login User*

Desain *interface* untuk *login user* yakni dengan memasukkan *username* dan *password*, jika *admin*, *user* ataupun kadis salah dalam memasukkan *username* dan *password* maka sistem menampilkan kembali ke halaman awal dengan masukkan *username* dan *password* kembali. Desain *interface* untuk *login user* disajikan pada Gambar 3.44.



**Gambar 3.44** Desain *Interface Login User Dan Admin*



### 3.4.5.3 Desain *Interface* Kelola Data Cuti Pegawai

Desain *interface* kelola data cuti pegawai adalah hak akses penuh yang diberikan kepada *admin*. Dalam *menu* ini *admin* dapat menghapus data cuti pegawai, mengedit, menambah, melihat jumlah data keseluruhan serta mencari data cuti dari masing-masing pegawai. Desain *interface* kelola data cuti pegawai disajikan pada Gambar 3.46.

The screenshot displays the user interface for managing employee leave data. At the top, the system title is 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA'. The navigation menu includes various administrative functions. The main content area shows a search bar and a table with the following structure:

No	Nama Pegawai	NIK	Tanggal Mulai Cuti	Tanggal Selesai Cuti	Keterangan	Aksi
						/
						/
						/
						/
						/
						/
						/
						/
						/
						/

**Gambar 3.46** Desain *Interface* Kelola Data Cuti Pegawai

### 3.4.5.4 Desain *Interface* Kelola Data Absensi Pegawai

Desain *interface* kelola data absensi pegawai adalah tugas yang diberikan kepada seorang *admin* agar dapat mengelola data absensi pegawai dengan baik. Pada *menu* absensi ini *admin* dapat menghapus data absensi pegawai, mengedit, menambah, melihat jumlah data keseluruhan absensi serta mencari data absensi dari masing-masing pegawai. Desain *interface* kelola data absensi pegawai disajikan pada Gambar 3.47.

The screenshot displays the user interface for the 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA'. The interface is primarily red and white. At the top, there is a header with the organization's logo on the left and its name in the center. On the right side of the header, there is a logo for 'JAYA RAYA'. Below the header, there is a navigation menu with several buttons: 'KELOLA DATA PEGAWAI', 'KELOLA DATA CUTI PEGAWAI', 'KELOLA DATA ABSENSI', 'KELOLA DATA PANGKAT PEGAWAI', 'KELOLA DATA PENDIDIKAN TERAKHIR', 'KELOLA DATA MUTASI PEGAWAI', 'KELOLA DATA PENSUNJUN PEGAWAI', 'KELOLA DATA FOTO BERKAS', 'KELOLA DATA GAJI PEGAWAI', 'MENGAWATI PASSWORD', 'MONITORING DATA', and 'CETAK LAPORAN'. Below the navigation menu, there is a section for 'Jumlah Data :-' and a '+ Input Data Absensi' button. To the right of this section is a search bar with a 'Search' button. The main content area is a table with the following columns: 'No', 'Nama Lengkap', 'NIP', 'Tanggal Absensi', 'Waktu', 'Keterangan (Terakhir/GMA)', and 'Aksi'. The 'Aksi' column contains edit and delete icons. The table is currently empty, showing only the header row.

No	Nama Lengkap	NIP	Tanggal Absensi	Waktu	Keterangan (Terakhir/GMA)	Aksi
						/
						/
						/
						/
						/
						/
						/
						/
						/
						/

**Gambar 3.47** Desain *Interface* Kelola Data Absensi Pegawai

### 3.4.5.5 Desain *Interface* Kelola Data Pangkat Pegawai

Desain *interface* kelola data pangkat pegawai adalah seorang *admin* yang diberikan tugas untuk mengelola seluruh data pangkat pegawai. Dalam *menu* ini *admin* mempunyai hak akses untuk dapat menghapus data pangkat pegawai, mengedit, menambah, melihat jumlah data pangkat secara keseluruhan serta mencari data pangkat pegawai dari masing-masing pegawai. Desain *interface* kelola data pangkat pegawai disajikan pada Gambar 3.48.

The screenshot displays the user interface for the 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA'. The interface is primarily red and white. At the top, there is a header with the organization's logo on the left and its name in the center. On the right side of the header, there is a logo for 'JAYA RAYA'. Below the header, there is a navigation menu with buttons for various administrative functions: 'KELOLA DATA PEGAWAI', 'KELOLA DATA CUTI PEGAWAI', 'KELOLA DATA ABSENSI', 'KELOLA DATA PANGKAT PEGAWAI', 'KELOLA DATA PENDIDIKAN TERAKHIR', 'KELOLA DATA MUTASI PEGAWAI', 'KELOLA DATA PENSUN PEGAWAI', 'KELOLA DATA FOTO BERKAS', 'KELOLA DATA GALI PEGAWAI', 'MENGGANTI PASSWORD', 'MONITORING DATA', and 'CETAK LAPORAN'. Below the navigation menu, there is a section for 'Jumlah Data :-' and a '+ Input Data Pangkat Pegawai' button. To the right of this section is a search bar with a 'Search' button. Below the search bar is a table with the following columns: 'No.', 'Nama Lengkap', 'NP', 'Golongan', 'Kelas', 'Kedudukan', and 'Aksi'. The 'Aksi' column contains edit and delete icons for each row. The table is currently empty.

No.	Nama Lengkap	NP	Golongan	Kelas	Kedudukan	Aksi
						/
						/
						/
						/
						/
						/
						/
						/
						/

**Gambar 3.48** Desain *Interface* Kelola Data Pangkat Pegawai

### 3.4.5.6 Desain *Interface* Kelola Data Pendidikan Terakhir

Desain *interface* kelola data pendidikan terakhir yang hak aksesnya diberikan penuh kepada *admin*, maka dalam *menu* ini *admin* dapat menghapus data pendidikan terakhir, mengedit, menambah, serta *menu* ini dapat menampilkan jumlah data keseluruhan pendidikan terakhir pegawai dan mencari data pendidikan terakhir dari masing-masing pegawai. Desain *interface* kelola data pendidikan terakhir disajikan pada Gambar 3.49.

The screenshot displays the user interface for the 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA'. The interface is primarily red and white. At the top, there is a header with the system name and logo on the left, and the 'JAYA RAYA' logo on the right. Below the header is a navigation menu with buttons for various administrative functions: 'KELOLA DATA PEGAWAI', 'KELOLA DATA CUTI PEGAWAI', 'KELOLA DATA ABSENSI', 'KELOLA DATA PANGKAT PEGAWAI', 'KELOLA DATA PENDIDIKAN TERAKHIR', 'KELOLA DATA MUTASI PEGAWAI', 'KELOLA DATA PENSUN PEGAWAI', 'KELOLA DATA FOTO BERKAS', 'KELOLA DATA BALI PEGAWAI', 'MENGGANTI PASSWORD', 'MONITORING DATA', and 'CETAK LAPORAN'. The main content area shows a search bar with the text 'Jumlah Data : -' and a '+ Input Data Pendidikan Pegawai' button. Below the search bar is a table with the following columns: 'No', 'Nama Lengkap', 'NIP', 'Pendidikan Terakhir', and 'Aksi'. The table contains several rows of data, each with a delete icon and an edit icon in the 'Aksi' column.

No	Nama Lengkap	NIP	Pendidikan Terakhir	Aksi
				/
				/
				/
				/
				/
				/
				/
				/
				/
				/

**Gambar 3.49** Desain *Interface* Kelola Data Pendidikan Terakhir



### 3.4.5.8 Desain *Interface* Kelola Data Pensiun Pegawai

Desain *interface* halaman kelola data pensiun pegawai dalam *menu* ini seorang *admin* diberikan hak akses penuh untuk dapat mengelola data pensiun pegawai dengan baik. Seorang *admin* pada *menu* ini dapat menghapus data pensiun pegawai, mengedit, menambah data, melihat jumlah data pensiun secara keseluruhan dan mencari data pensiun dari masing-masing pegawai. Desain *interface* kelola data pensiun pegawai disajikan pada Gambar 3.51.

The screenshot displays the user interface for the 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA'. The interface is primarily red and white. At the top, there is a header with the system name and logos for 'DISPORA' and 'JAYA RAYA'. Below the header is a navigation menu with buttons for various administrative functions: KELOLA DATA PEGAWAI, KELOLA DATA CUTI PEGAWAI, KELOLA DATA ABSENSI, KELOLA DATA PANGKAT PEGAWAI, KELOLA DATA PENDIDIKAN TERAKHIR, KELOLA DATA MUTASI PEGAWAI, KELOLA DATA PENSUN PEGAWAI, KELOLA DATA FOTO BERKAS, KELOLA DATA GAJI PEGAWAI, MENGGANTI PASSWORD, MONITORING DATA, and CETAK LAPORAN. The main content area shows a table for managing pension data. The table has columns for No, Nama Jengkep, NP, Tahun Pensiun, Golongan, Kelas, Kedudukan, Gaji, and Pensiun, along with an 'Aksi' column containing edit and delete icons. Above the table, there is a search bar and a button labeled 'Input Data Pensiun Pegawai'.

No	Nama Jengkep	NP	Tahun Pensiun	Golongan	Kelas	Kedudukan	Gaji	Pensiun	Aksi
									/
									/
									/
									/
									/
									/
									/
									/
									/
									/

**Gambar 3.51** Desain *Interface* Kelola Data Pensiun Pegawai

### 3.4.5.9 Desain *Interface* Mengganti *Password*

Desain *interface* halaman untuk mengganti *password* adalah hak akses penuh yang diberikan kepada seorang *admin*. Gambar dibawah ini yang ditampilkan oleh sistem hanya *admin* yang dapat mengaturnya. Dalam menu ini *admin* dapat mencari *username*, serta mengganti *password* dari masing-masing pegawai bila pegawai tidak ingat dengan *password* ataupun salah memasukkan *password* yang telah digunakan sebelumnya. Desain *interface* untuk *menu* mengganti *password* disajikan pada Gambar 3.52.

The image shows a web application interface for the 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA'. The interface is primarily red and white. At the top, there is a header with the DISPORA logo on the left and the system title in the center. On the right side of the header, there is a logo for 'JAYA RAYA'. Below the header is a navigation menu with several buttons: 'KELOLA DATA PEGAWAI', 'KELOLA DATA CUTI PEGAWAI', 'KELOLA DATA ABSENSI', 'KELOLA DATA PANGKAT PEGAWAI', 'KELOLA DATA PENDIKIHAN TERAKHIR', 'KELOLA DATA MUTASI PEGAWAI', 'KELOLA DATA PENSILIH PEGAWAI', 'KELOLA DATA FOTO BERKAS', 'KELOLA DATA GALI PEGAWAI', 'MENGANTI PASSWORD', 'MONITORING DATA', and 'CETAK LAPORAN'. The 'MENGANTI PASSWORD' button is highlighted. Below the navigation menu, there is a search bar with the placeholder text 'Masukkan usemame' and a 'Search' button. Below the search bar, the text 'Hasil pencarian :' is displayed. There are two main panels: 'User Terdaftar' and 'Ganti Password'. The 'User Terdaftar' panel has two input fields: 'Username' and 'Password Lama'. The 'Ganti Password' panel has two input fields: 'Password Baru' and 'Konfirmasi Password Baru'. A 'Simpan' button is located at the bottom right of the 'Ganti Password' panel.

**Gambar 3.52** Desain *Interface* Mengganti *Password*

### 3.4.5.10 Desain *Interface Monitoring Data*

Desain *interface monitoring data* seorang *admin* dalam *menu* ini mempunyai hak akses penuh atas *menu monitoring data*. Pada *menu monitoring data* ini seorang *admin* dapat melihat jumlah data yang ditampilkan oleh sistem secara keseluruhan dengan pengelompokkan dari masing-masing *menu* yang hanya dikelola oleh seorang *admin*, serta dalam *menu* ini sistem menampilkan tanggal akses terakhir dari masing-masing data. Desain *interface monitoring data* disajikan pada Gambar 3.53.

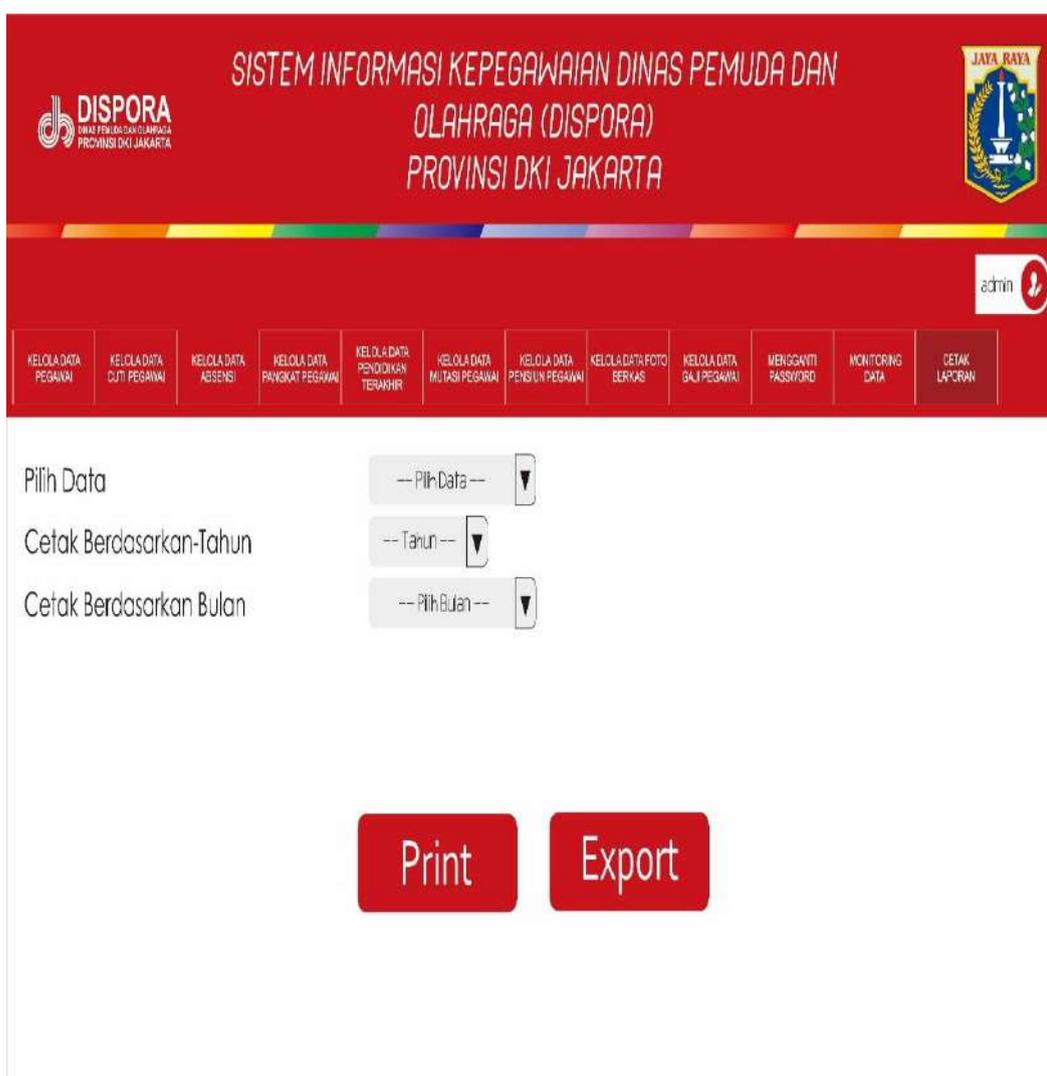
The screenshot displays the monitoring data interface for the 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA'. The interface features a navigation menu with 12 items and a main content area with 8 data cards. Each card displays the category name, the number of data items, and the last access date.

Menu	Jumlah Data	Tanggal Akses Terakhir
KELOLA DATA PEGAWAI	100	15-2-2017
KELOLA DATA PENDIDIKAN TERAKHIR	9	21-2-2017
KELOLA DATA CUTI PEGAWAI	7	12-2-2017
KELOLA DATA MUTASI PEGAWAI	2	2-2-2017
KELOLA DATA ABSENSI	3	17-2-2017
KELOLA DATA PENSIUN PEGAWAI	13	24-2-2017
KELOLA DATA PANGKAT PEGAWAI	16	10-2-2017
KELOLA DATA FOTO	152	24-3-2017

**Gambar 3.53** Desain *Interface Monitoring Data*

### 3.4.5.11 Desain *Interface* Cetak Laporan

Desain *interface* cetak laporan yakni dimana seorang *admin* diberikan hak akses penuh untuk *menu* cetak laporan. Didalam *menu* ini *admin* dapat mencetak laporan dengan memilih masing-masing data yang telah disajikan oleh dalam sistem, setelah itu *admin* juga dapat memilih tahun serta bulan yang diinginkan. Kemudian *menu* juga menyediakan *pengekspor* laporan untuk mempermudah *admin* dalam mencetak laporan, dalam *menu* ini sistem menyediakan dua pilihan untuk *mengekspor* laporan dapat menggunakan *Microsoft Excel* atau pun bentuk *file PDF*. Desain *interface* cetak laporan disajikan pada Gambar 3.54.



**Gambar 3.54** Desain *Interface* Cetak Laporan

### 3.4.5.12 Desain *Interface* Kelola Data Foto

Desain *interface* kelola data foto dalam *menu* ini seorang *admin* diberikan hak akses penuh untuk dapat mengelola data foto. Pada bagian ini *admin* dapat menghapus data foto, mengedit atau memperbaharui, menambah, melihat jumlah data foto secara keseluruhan serta mencari data foto dari masing-masing pegawai. Desain *interface* kelola data foto disajikan pada Gambar 3.55.

The screenshot displays the user interface for managing photo data. At the top, the header includes the logo of DISPORA (Dinas Pemuda dan Olahraga Provinsi DKI Jakarta) and the system title: "SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLARAGA (DISPORA) PROVINSI DKI JAKARTA". A user profile for "admin" is visible in the top right corner. The main navigation bar contains the following menu items: KELOLA DATA PEGAWAI, KELOLA DATA OUTI PEGAWAI, KELOLA DATA ABSENSI, KELOLA DATA PANGKAT PEGAWAI, KELOLA DATA PENDIDIKAN TERAKHIR, KELOLA DATA MUTASI PEGAWAI, KELOLA DATA PENSUN PEGAWAI, KELOLA DATA FOTO BERKAS, KELOLA DATA GAJI PEGAWAI, MENGANTI PASSWORD, MONITORING DATA, and CETAK LAPORAN. Below the navigation bar, there is a search bar and a table with the following columns: No., Nama Lengkap, NP, File Foto Berkas, and Aksi. The table contains several rows of data, each with a trash icon in the Aksi column.

No.	Nama Lengkap	NP	File Foto Berkas	Aksi
				/
				/
				/
				/
				/
				/
				/
				/
				/

**Gambar 3.55** Desain *Interface* Kelola Data Foto

### 3.4.6 Rancangan *Input* untuk Akses Pegawai

Rancangan *input* data untuk akses pegawai yang digunakan oleh seorang pegawai agar dapat menginput data pegawai, menginput data pengajuan cuti, menginput absen, menginput foto, menginput data pensiun, menginput data mutasi. Berikut ini adalah desain *interface* pada rancangan *input* untuk akses pegawai :

#### 3.4.6.1 Desain *Interface Input* Data Pengajuan Cuti

Desain *interface input* data pengajuan cuti adalah akses yang diberikan dari *admin* kepada pegawai agar seorang pegawai dapat memasukkan data cuti yang dapat diambil oleh pegawai tersebut. Pegawai hanya dapat melihat data pengajuan cuti yang pernah diambil, menyimpan data pengajuan cuti, serta mengisi *form* yang telah tersedia. Desain *interface input* data pengajuan cuti disajikan pada Gambar 3.56.

The image shows a web application interface for 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLARHAGA (DISPORA) PROVINSI DKI JAKARTA'. The main navigation bar contains several menu items: 'INPUT DATA PEGAWAI', 'INPUT DATA ABSENSI PEGAWAI', 'INPUT DATA MUTASI PEGAWAI', 'INPUT FOTO', 'INPUT DATA PENSUN', and 'INPUT DATA PENGAJUAN CUTI PEGAWAI'. The 'INPUT DATA PENGAJUAN CUTI PEGAWAI' menu is highlighted. Below the navigation bar, the 'Data Pengajuan Cuti' form is displayed. The form has a red header and contains the following fields: 'Nama' (text input), 'NIP' (text input), 'Tanggal Mulai Cuti' (date input), 'Tanggal Selesai Cuti' (date input), and 'Keterangan' (text area). At the bottom of the form, there are two buttons: 'Simpan' and 'Lihat'.

**Gambar 3.56** Desain *Interface Input* Data Pengajuan Cuti

### 3.4.6.2 Desain *Interface Input Absen*

Desain *interface input* absen yakni akses yang diberikan kepada pegawai dari *admin* agar pegawai dapat menginputkan absen oleh masing-masing pegawai sesuai dengan keperluan. Dalam *menu* absen ini pegawai hanya dapat menyimpan data absen, pegawai mengisi *form* yang telah tersedia dalam *menu* ini serta memasukkan absen dengan keterangan sebagai tanda bukti. Desain *interface input* absen disajikan pada Gambar 3.57.

The image shows a screenshot of a web application interface for the 'Absen' (Absence) module. The interface is set against a red background with white text and elements. At the top, the header reads 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLARAGA (DISPORA) PROVINSI DKI JAKARTA'. On the left is the DISPORA logo, and on the right is the logo of the Province of Jakarta. Below the header is a navigation bar with several menu items: 'INPUT DATA PEGAWAI', 'INPUT DATA ABSENSI PEGAWAI', 'INPUT DATA MUTASI PEGAWAI', 'INPUT FOTO', 'INPUT DATA PENSUN', and 'INPUT DATA PENGAJUAN CUTI PEGAWAI'. The 'INPUT DATA ABSENSI PEGAWAI' menu is highlighted. The main content area features a form titled 'Absen' with the following fields: 'Nama' (text input), 'NIP' (text input), 'Tanggal Absen' (date input), 'Waktu' (text input), and 'Keterangan' (text input with a dropdown menu for 'Pilih Keterangan'). At the bottom of the form are two buttons: 'Simpan' (Save) and 'Lihat' (View).

**Gambar 3.57** Desain *Interface Input Absen*

### 3.4.6.3 Desain *Interface Input Foto*

Desain *interface input* foto yakni pegawai diberikan akses oleh *admin* untuk dapat menginputkan *file* berupa foto. Dalam *menu* ini seorang pegawai dapat menyimpan foto, melihat serta memasukkan foto dalam bentuk nama *file* atau *blob* dengan mengisi *form* yang telah tersedia dalam *menu* ini. Desain *interface input* foto disajikan pada Gambar 3.58.



The image shows a web application interface for 'SISTEM INFORMASI KEPEGAWAIAN' (Employee Information System) of the 'DINAS PEMUDA DAN OLARAGA (DISPORA) PROVINSI DKI JAKARTA'. The interface is primarily red and white. At the top, there is a navigation bar with several menu items: 'INPUT DATA PEGAWAI', 'INPUT DATA ABSENSI PEGAWAI', 'INPUT DATA MUTASI PEGAWAI', 'INPUT FOTO' (which is highlighted), 'INPUT DATA PENSUN', and 'INPUT DATA PENGALUAN CUTI PEGAWAI'. A 'Pegawai' profile icon is also visible in the top right. The main content area is titled 'Foto' and contains a form with three input fields: 'Nama', 'NIP', and 'Foto'. The 'Foto' field has a 'Browse...' button next to it. Below the form are two buttons: 'Simpan' (Save) and 'Lihat' (View).

**Gambar 3.58** Desain *Interface Input Foto*

### 3.4.6.4 Desain *Interface Input Data Pegawai*

Desain *interface input* data pegawai adalah akses yang diberikan *admin* kepada pegawai untuk *menu input* data pegawai ini, agar seorang pegawai dapat memasukkan data diri oleh masing-masing pegawai. Dalam *menu input* data pegawai seorang pegawai hanya dapat menyimpan data pegawai, serta melihat data pegawai, sebelum data disimpan kedalam *database*, pegawai dapat mengisi *form* yang telah disediakan dalam *menu* ini. Berikutnya pegawai juga dapat menyertakan *file* berupa foto yang diperlukan dalam pengisian data pegawai. Desain *interface input* data pegawai disajikan pada Gambar 3.59.

The image shows a web application interface for a personnel information system. The header is red and contains the logo of 'DISPORA' (Dinas Pemuda dan Olahraga Provinsi DKI Jakarta) and the text 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLARAGA (DISPORA) PROVINSI DKI JAKARTA'. Below the header is a navigation bar with several buttons: 'INPUT DATA PEGAWAI', 'INPUT DATA ABSSENSI PEGAWAI', 'INPUT DATA MUTASI PEGAWAI', 'INPUT FOTO', 'INPUT DATA PENSIUN', and 'INPUT DATA PENGAJUAN CUTI PEGAWAI'. The main content area is divided into two sections: 'Daftar Riwayat Hidup' and 'Keterangan Lain'. The 'Daftar Riwayat Hidup' section contains various input fields and dropdown menus for personal information such as 'Tipe PNS', 'NIP', 'Nama', 'Tempat & Tanggal Lahir', 'Jenis Kelamin', 'Agama', 'Status Nikah', 'Kedudukan PNS', and 'Golongan Darah'. The 'Keterangan Lain' section includes fields for 'Kegiatan/Hobi', 'No. KARPEG', 'No. AKSES', 'No. TASPEN', 'No. NPWP', 'No. Kartu Kedis', 'NIK', and 'Foto/Berkas Pegawai'. At the bottom right, there are 'Simpan' and 'Lihat' buttons.

**Gambar 3.59** Desain *Interface Input Data Pegawai*

### 3.4.6.5 Desain *Interface Input Data Pensiun*

Desain *interface input* data pensiun yang digambarkan dibawah ini adalah akses yang diberikan *admin* kepada pegawai, seorang pegawai dapat memasukkan data pensiunnya sesuai dengan aturan yang berlaku. Pada *menu input* data pensiun pegawai ini seorang pegawai dapat menyimpan data pensiun dengan mengisi data pensiun terlebih dahulu yakni mengisi *form* yang telah disediakan dalam *menu* ini, serta pegawai dapat melihat data pensiunnya sendiri. Desain *interface input* data pensiun disajikan pada Gambar 3.60.

The image shows a web interface for the 'Data Pensiun' (Pension Data) form. The header includes the logo of DISPORA (DINAS PEMUDA DAN OLARAGA) PROVINSI DKI JAKARTA and the official emblem of the province. A navigation bar contains several menu items: 'INPUT DATA PEGAWAI', 'INPUT DATA ABSENSI PEGAWAI', 'INPUT DATA MUTASI PEGAWAI', 'INPUT FOTO', 'INPUT DATA PENSUN', and 'INPUT DATA PENGALUAN CUTI PEGAWAI'. The 'INPUT DATA PENSUN' menu is currently selected. The main content area displays the 'Data Pensiun' form with the following fields:

- Nama: Text input field
- NIP: Text input field
- Tahun Pensiun: Dropdown menu with 'Pilih Tahun' as the placeholder
- Golongan: Dropdown menu with 'Pilih Golongan' as the placeholder
- Kelas: Dropdown menu with 'Pilih Kelas' as the placeholder
- Kedudukan: Dropdown menu with 'Pilih Kedudukan' as the placeholder
- Gaji: Text input field
- Tunjangan: Text input field

At the bottom of the form, there are two buttons: 'Simpan' (Save) and 'Lihat' (View).

**Gambar 3.60** Desain *Interface Input Data Pensiun*

### 3.4.6.6 Desain *Interface Input Data Mutasi Pegawai*

Desain *interface input* data mutasi adalah akses *menu* untuk memasukkan data mutasi pegawai yang diberikan *admin* kepada pegawai agar pegawai dapat memasukkan data mutasi sesuai dengan kebutuhan. Pada *menu* ini pegawai dapat menyimpan data mutasi dengan memasukkan data terlebih dahulu menggunakan *form* yang telah tersedia dalam *menu* ini, pegawai juga dapat melihat data mutasi bila sebelumnya sudah ada data mutasi dari pegawai tersebut. Desain *interface input* data mutasi pegawai dapat diperhatikan dalam Gambar 3.61.

The image shows a web interface for the 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA'. The main header is red with the organization's logo and name. Below the header is a navigation bar with several menu items: 'INPUT DATA PEGAWAI', 'INPUT DATA ABSSENSI PEGAWAI', 'INPUT DATA MUTASI PEGAWAI', 'INPUT FOTO', 'INPUT DATA PENSUN', and 'INPUT DATA PENGAJUAN CUTI PEGAWAI'. The 'INPUT DATA MUTASI PEGAWAI' menu is highlighted. The main content area displays a form titled 'Data Mutasi' with the following fields:

- No Surat:
- Penhal:
- Nama:
- NIP:
- Tanggal Mutasi:
- Tanggal Masuk:
- Jabatan Lama:
- Jabatan Baru:
- Golongan:
- Kelas:
- Kedudukan:

At the bottom of the form are two buttons: 'Simpan' and 'Lihat'.

**Gambar 3.61** Desain *Interface Input Data Mutasi Pegawai*

### 3.4.7 Rancangan untuk Akses Kepala Dinas

Rancangan untuk akses kepala dinas yang digunakan oleh seorang kadis agar dapat memverifikasi berkas pengajuan pegawai, memverifikasi berkas pengajuan cuti, memverifikasi berkas pengajuan pensiun, memverifikasi berkas mutasi, serta monitoring data pegawai. Berikut ini adalah desain *interface* pada rancangan untuk akses kepala dinas :

#### 3.4.7.1 Desain *Interface Verifikasi Berkas Pengajuan Cuti*

Desain *interface verifikasi* berkas pengajuan cuti adalah akses yang diberikan dari *admin* kepada kadis agar seorang kadis dapat memverifikasi berkas pengajuan cuti yang dapat diambil oleh pegawai. Kadis hanya dapat memverifikasi berkas pengajuan cuti yang diajukan oleh pegawai. Desain *interface verifikasi* berkas pengajuan cuti disajikan pada Gambar 3.62.



**Gambar 3.62** Desain *Interface Verifikasi Berkas Pengajuan Cuti*

### 3.4.7.2 Desain *Interface Verifikasi Berkas Pengajuan Mutasi*

Desain *interface verifikasi* berkas pengajuan mutasi adalah akses yang diberikan dari *admin* kepada kadis agar seorang kadis dapat memverifikasi berkas pengajuan mutasi yang dapat diambil oleh pegawai. Kadis hanya dapat memverifikasi berkas pengajuan mutasi yang diajukan oleh pegawai. Desain *interface verifikasi* berkas pengajuan mutasi disajikan pada Gambar 3.63.

The screenshot displays the user interface for the 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA'. The interface is primarily red and white. At the top, there is a header with the organization's name and logo. Below the header, there are four main menu items: 'VERIFIKASI BERKAS CUTI', 'VERIFIKASI BERKAS MUTASI' (highlighted), 'VERIFIKASI BERKAS Pensiun', and 'MONITORING DATA'. A search bar is located above a table. The table has columns for 'No', 'Nama Lengkap', 'Pangkat / Gol', 'Tanggal Surat Masuk', 'Tajuk Mutasi', 'Keterangan', and 'Aksi'. The 'Aksi' column contains a dropdown arrow and a 'KADIS' button for each row.

No	Nama Lengkap	Pangkat / Gol	Tanggal Surat Masuk	Tajuk Mutasi	Keterangan	Aksi
						⌵ KADIS
						⌵ KADIS
						⌵ KADIS
						⌵ KADIS
						⌵ KADIS
						⌵ KADIS
						⌵ KADIS
						⌵ KADIS
						⌵ KADIS
						⌵ KADIS

**Gambar 3.63** Desain *Interface Verifikasi Berkas Pengajuan Mutasi*

### 3.4.7.3 Desain *Interface Verifikasi Berkas Pengajuan Pensiun*

Desain *interface verifikasi* berkas pengajuan pensiun adalah akses yang diberikan dari *admin* kepada kadis agar seorang kadis dapat memverifikasi berkas pengajuan pensiun yang dapat diambil oleh pegawai. Kadis hanya dapat memverifikasi berkas pengajuan pensiun yang diajukan oleh pegawai. Desain *interface verifikasi* berkas pengajuan pensiun disajikan pada Gambar 3.64.

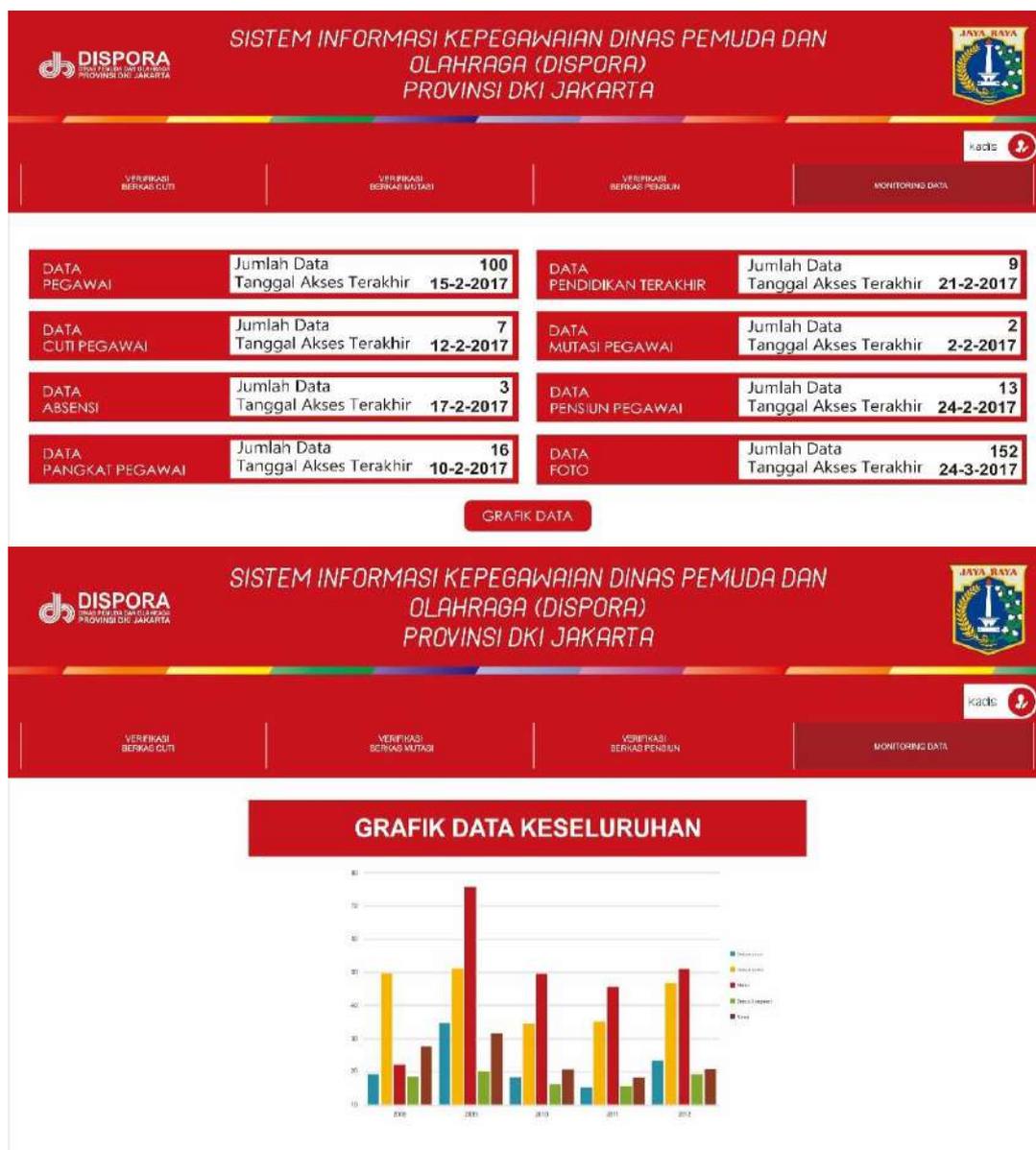
The screenshot displays the user interface for verifying pension application files. At the top, the system title is 'SISTEM INFORMASI KEPEGAWAIAN DINAS PEMUDA DAN OLAHRAGA (DISPORA) PROVINSI DKI JAKARTA'. The navigation menu includes options for 'VERIFIKASI BERKAS CUTI', 'VERIFIKASI BERKAS MUTASI', 'VERIFIKASI BERKAS PENSUN', and 'MONITORING DATA'. Below the navigation, there is a search bar and a table listing application data. The table has the following columns: No, Nama Lengkap, Pangkat / Gol, Tanggal Surat Masuk, Jumlah Mutasi, Keterangan, and Aksi. Each row in the table represents an application and includes a 'Verifikasi' button and a 'Kadid' icon in the 'Aksi' column.

No	Nama Lengkap	Pangkat / Gol	Tanggal Surat Masuk	Jumlah Mutasi	Keterangan	Aksi
						Verifikasi Kadid
						Verifikasi Kadid
						Verifikasi Kadid
						Verifikasi Kadid
						Verifikasi Kadid
						Verifikasi Kadid
						Verifikasi Kadid
						Verifikasi Kadid
						Verifikasi Kadid
						Verifikasi Kadid

**Gambar 3.64** Desain *Interface Verifikasi Berkas Pengajuan Pensiun*

### 3.3.7.4 Desain *Interface Monitoring Data*

Desain *interface input monitoring data* adalah akses *menu* yang diberikan admin kepada kepala dinas dalam menu ini kadis hanya dapat melihat data-data pegawai sesuai dengan kebutuhan, dalam *menu* ini kadis tidak dapat mengedit, serta menghapus data. Kepala dinas juga dapat melihat perkembangan data melalui grafik data. Desain *interface monitoring data* pegawai dapat diperhatikan dalam Gambar 3.65.



**Gambar 3.65** Desain *Interface Monitoring Data* Dan Grafik Data

### 3.5 Penulisan Kode Program

Penulisan kode program menggunakan bahasa PHP, HTML, CSS, dan *javascript* menggunakan *framework* Laravel 5.6.

#### 3.5.1 Pengujian Program

Pada penelitian ini dilakukan pengujian sistem menggunakan metode *equivalence partitioning*. Pengujian sistem ini dilakukan untuk menguji fungsi keseluruhan sistem (fungsionalitas) apakah berjalan dengan baik, dapat mengetahui kesalahan yang terjadi dan kesalahan struktur data maupun *interface*. Rancangan daftar pengujian disajikan pada Tabel 3.36, Tabel 3.37 dan Tabel 3.38.

Tabel 3.36. Daftar Pengujian pada Admin

NO	KELAS UJI	DAFTAR PENGUJIAN	KASUS UJI	HASIL YANG DIHARAPKAN
1	Menu Pegawai	Fungsional	Mengakses	Halaman dapat
		Tampilan Data Pegawai	tampilan halaman data pegawai	Menampilkan data pegawai yang telah diisi
		Fungsional Tampilan Tambah Data Pegawai	Mengakses halaman tambah data pegawai	Halaman dapat menambahkan data pegawai

Tabel 3.36. Daftar Pengujian pada Admin (Lanjutan)

NO	KELAS UJI	DAFTAR PENGUJIAN	KASUS UJI	HASIL YANG DIHARAPKAN
1	Menu	Fungsional	Mengakses	Halaman dapat
	Pegawai	Tampilan	Tampilan	menghapus
		Hapus Data Pegawai	Halaman hapus data pegawai	data pegawai
		Fungsional Tampilan Edit Data Pegawai	Mengakses tampilan halaman edit data Pegawai	Halaman dapat memperbaharui data pegawai
		Fungsional Tampilan Lihat Data Pegawai	Mengakses Tampilan Halaman lihat data Pegawai	Menampilkan detail data Pegawai
	Fungsional Tampilan Cari Data Pegawai	Mengakses Tampilan Halaman cari data Pegawai	Halaman dapat menampilkan data yang dicari	
2	Menu Cuti	Fungsional Tampilan	Mengakses tampilan	Halaman dapat menampilkan
		Halaman Data Cuti	halaman data Cuti	data cuti yang telah diisi
		Fungsional Tampilan Halaman Tambah Data Cuti	Mengakses Tampilan Halaman tambah data Cuti	Halaman dapat menambahkan data cuti
		Fungsional Tampilan Lihat Data Cuti	Mengakses Tampilan Halaman lihat data cuti	Menampilkan detail data cuti

Tabel 3.36. Daftar Pengujian pada Admin (Lanjutan)

NO	KELAS UJI	DAFTAR PENGUJIAN	KASUS UJI	HASIL YANG DIHARAPKAN
2	Menu Cuti	Fungsional Tampilan Hapus Cuti	Mengakses tampilan halaman hapus data cuti	Halaman dapat menghapus data cuti
		Fungsional Tampilan Halaman Edit Data Cuti	Mengakses tampilan halaman edit data cuti	Halaman dapat memperbaharui data cuti
		Fungsional Tampilan Halaman Cari Data Cuti	Mengakses tampilan halaman cari data cuti	Halaman dapat menampilkan data yang dicari
3	Menu Absen	Fungsional Tampilan Halaman Data Absen	Mengakses tampilan halaman data Absen	Halaman dapat menampilkan data absen yang telah diisi
		Fungsional Tampilan Tambah Data Absen	Mengakses tampilan halaman tambah data Absen	Halaman dapat menambahkan data absen
		Fungsional Tampilan Edit Data Absen	Mengakses tampilan halaman edit data Absen	Halaman dapat memperbaharui data absen
		Fungsional Tampilan Hapus Data Absen	Mengakses tampilan halaman hapus data absen	Halaman dapat menghapus data absen

Tabel 3.36. Daftar Pengujian pada Admin (Lanjutan)

NO	KELAS UJI	DAFTAR PENGUJIAN	KASUS UJI	HASIL YANG DIHARAPKAN
3	Menu Absen	Fungsional Tampilan Lihat Data Absen	Mengakses tampilan halaman lihat data absen	Menampilkan detail data absen
		Fungsional Tampilan Cari Data Absen	Mengakses tampilan halaman cari data absen	Halaman dapat menampilkan data yang dicari
4	Menu Pangkat	Fungsional Tampilan Halaman Data Pangkat	Mengakses Tampilan Halaman data pangkat	Halaman dapat menampilkan data pangkat yang telah diisi
		Fungsional Tampilan Tambah Data Pangkat	Mengakses tampilan halaman tambah data pangkat	Halaman dapat menambahkan data pangkat
		Fungsional Tampilan Edit Data Pangkat	Mengakses tampilan halaman edit data	Halaman dapat memperbaharui data pangkat
		Fungsional Tampilan Hapus Data Pangkat	Mengakses tampilan halaman hapus data pangkat	Halaman dapat menghapus data pangkat
		Fungsional Tampilan Lihat Data Pangkat	Mengakses tampilan halaman lihat data pangkat	Menampilkan detail data pangkat

Tabel 3.36. Daftar Pengujian pada Admin (Lanjutan)

NO	KELAS UJI	DAFTAR PENGUJIAN	KASUS UJI	HASIL YANG DIHARAPKAN
4	Menu Pangkat	Fungsional Tampilan Halaman Cari Data Pangkat	Mengakses tampilan halaman cari data pangkat	Halaman dapat menampilkan data yang dicari
5	Menu Pendidikan	Fungsional Tampilan Halaman Data Pendidikan	Mengakses tampilan halaman data Pendidikan	Halaman dapat menampilkan data pendidikan yang telah diisi
		Fungsional Tampilan Halaman Tambah Data Pendidikan	Mengakses tampilan halaman tambah data Pendidikan	Halaman dapat menambahkan data pendidikan
		Fungsional Tampilan Halaman Hapus Data Pendidikan	Mengakses Tampilan Halaman hapus data Pendidikan	Halaman dapat menghapus data pendidikan
		Fungsional Tampilan Halaman Edit Data Pendidikan	Mengakses tampilan halaman edit data pendidikan	Halaman dapat memperbaharui data pendidikan
		Fungsional Tampilan Halaman Lihat Data Pendidikan	Mengakses tampilan halaman lihat data Pendidikan	Menampilkan detail data Pendidikan

Tabel 3.36. Daftar Pengujian pada Admin (Lanjutan)

NO	KELAS UJI	DAFTAR PENGUJIAN	KASUS UJI	HASIL YANG DIHARAPKAN
5	Menu Pendidikan	Fungsional Tampilan Halaman Cari Data Pendidikan	Mengakses tampilan halaman cari data Pendidikan	Halaman dapat menampilk data yang dicari
6	Menu Mutasi	Fungsional Tampilan Halaman Data Mutasi	Mengakses tampilan halaman data mutasi	Halaman dapat menampilkan data mutasi yang telah diisi
		Fungsional Tampilan Halaman Tambah Data Mutasi	Mengakses tampilan halaman tambah data mutasi	Halaman dapat menambah data mutasi
		Fungsional Tampilan Halaman Hapus Data Mutasi	Mengakses tampilan halaman hapus data mutasi	Halaman dapat menghapus data mutasi
		Fungsional Tampilan Halaman Edit Data Mutasi	Mengakses tampilan halaman edit data mutasi	Halaman dapat memperbaharui data mutasi
		Fungsional Tampilan Halaman Lihat Data Mutasi	Mengakses tampilan halaman lihat data mutasi	Menampilkan detail data mutasi
		Fungsional Tampilan Halaman Cari Data Mutasi	Mengakses tampilan halaman cari data mutasi	Halaman dapat menampilkan data yang dicari

Tabel 3.36. Daftar Pengujian pada Admin (Lanjutan)

NO	KELAS UJI	DAFTAR PENGUJIAN	KASUS UJI	HASIL YANG DIHARAPKAN
7	Menu	Fungsional	Mengakses	Halaman dapat
	Pensiun	Tampilan Halaman Data Pensiun	Tampilan halaman data pensiun	menampilkan data pensiun yang telah diisi
		Fungsional Tampilan Halaman Tambah Data Pensiun	Mengakses tampilan halaman tambah data Pensiun	Halaman dapat menampilkan data pensiun
		Fungsional Tampilan Halaman Hapus Data Pensiun	Mengakses tampilan halaman hapus data pensiun	Halaman dapat menghapus data pensiun
		Fungsional Tampilan Halaman Edit Data Pensiun	Mengakses tampilan halaman edit data pensiun	Halaman dapat memperbaharui data pensiun
		Fungsional Tampilan Halaman Lihat Data Pensiun	Mengakses tampilan halaman lihat data pensiun	Menampilkan detail data pensiun
		Fungsional Tampilan Halaman Cari Data Pensiun	Mengakses tampilan halaman cari data pensiun	Halaman dapat menampilkan data yang dicari

Tabel 3.36. Daftar Pengujian pada Admin (Lanjutan)

NO	KELAS UJI	DAFTAR PENGUJIAN	KASUS UJI	HASIL YANG DIHARAPKAN
8	Menu	Fungsional	Mengakses	Halaman dapat
	Foto	Tampilan Halaman Data Foto	tampilan Halaman data foto	menampilkan data pensiun yang telah diisi
		Fungsional Tampilan Halaman Tambah Data Foto	Mengakses tampilan halaman tambah data foto	Halaman dapat menambahkan data foto
		Fungsional Tampilan Halaman Hapus Data Foto	Mengakses tampilan halaman hapus data foto	Halaman dapat menghapus data foto
		Fungsional Tampilan Halaman Edit Data Foto	Mengakses tampilan halaman edit data foto	Halaman dapat memperbaharui data foto
		Fungsional Tampilan Halaman Lihat Data Foto	Mengakses tampilan halaman lihat data foto	Menampilkan detail data foto
		Fungsional Tampilan Halaman Cari Data Foto	Mengakses tampilan halaman cari data foto	Halaman dapat menampilkan data yang dicari

Tabel 3.36. Daftar Pengujian pada Admin (Lanjutan)

NO	KELAS UJI	DAFTAR PENGUJIAN	KASUS UJI	HASIL YANG DIHARAPKAN
9	Menu	Fungsional	Mengakses	Menampilkan
	Ganti Password	Tampilan Halaman Ganti Password	Tampilan halaman ganti password	halaman ganti password
		Fungsional Tampilan Halaman Cari Username	Mengakses tampilan halaman cari username	Halaman dapat menampilkan username terdaftar
		Fungsional Tampilan Halaman User Terdaftar	Mengakses tampilan halaman user terdaftar	Halaman dapat menampilkan user yang telah terdaftar
		Fungsional Tampilan Halaman Ganti Password	Mengakses tampilan halaman ganti password	Halaman dapat memperbaharui password baru
		Fungsional Tampilan Halaman Simpan Password	Mengakses tampilan halaman simpan password	Halaman dapat menyimpan password baru

Tabel 3.36. Daftar Pengujian pada Admin (Lanjutan)

NO	KELAS UJI	DAFTAR PENGUJIAN	KASUS UJI	HASIL YANG DIHARAPKAN
10	Menu	Fungsional	Mengakses	Halaman dapat
	Monitoring Data	Tampilan Halaman Monitoring Data	tampilan halaman monitoring data	menampilkan seluruh data yang telah diisi
		Fungsional Tampilan Halaman Graik Data	Mengakses tampilan halaman grafik data	Halaman dapat menampilkan grafik keseluruhan data
11	Menu Cetak Laporan	Fungsional Tampilan Halaman Cetak Laporan	Mengakses tampilan halaman cetak laporan	Menampilkan halaman cetak laporan
		Fungsional Tampilan Halaman Pilih Data	Mengakses tampilan halaman pilih data	Halaman dapat memilih data
		Fungsional Tampilan Halaman Cetak Data Per tahun/ Bulan	Mengakses tampilan halaman cetak data per tahun/bulan	Halaman dapat memilih data berdasarkan tahun/bulan
		Fungsional Tampilan Halaman Print	Mengakses tampilan halaman print	Halaman dapat menampilkan data yang telah dicetak
		Fungsional Tampilan Halaman Export	Mengakses tampilan halaman export	Halaman dapat menampilkan pilihan cetak data format pdf atau excel

Tabel 3.37 Daftar Pengujian pada Pegawai

<b>NO</b>	<b>KELAS UJI</b>	<b>DAFTAR PENGUJIAN</b>	<b>KASUS UJI</b>	<b>HASIL YANG DIHARAPKAN</b>
1	Menu Input Data Pegawai	Fungsional Tampilan Halaman Input Data Pegawai	Mengakses tampilan halaman input data pegawai	Halaman dapat menampilkan form input Pegawai
		Fungsional Tampilan Halaman Input Data	Mengakses tampilan halaman input data	Data dapat diisi dan dapat disimpan
		Fungsional Tampilan Halaman Simpan Data	Mengakses tampilan halaman simpan data	Halaman dapat menyimpan data pegawai
		Fungsional Tampilan Halaman Lihat Data Pegawai	Mengakses tampilan halaman lihat data pegawai	Halaman dapat menampilkan data pegawai tersimpan
2	Menu Input Absen	Fungsional Tampilan Halaman Input Absen	Mengakses tampilan halaman input	Halaman dapat menampilkan form input absen
		Fungsional Tampilan Halaman Input Absen	Mengakses tampilan halaman input absen	Data dapat diisi dan dapat disimpan
		Fungsional Tampilan Halaman Simpan Absen	Mengakses tampilan halaman simpan absen	Halaman dapat menyimpan data Absen

Tabel 3.37 Daftar Pengujian pada Pegawai (Lanjutan)

<b>NO</b>	<b>KELAS UJI</b>	<b>DAFTAR PENGUJIAN</b>	<b>KASUS UJI</b>	<b>HASIL YANG DIHARAPKAN</b>
2	Menu Input Absen	Fungsional Tampilan Halaman Lihat Absen	Mengakses tampilan halaman lihat absen	Halaman dapat menampilkan data absen pegawai tersimpan
3	Menu Input Foto	Fungsional Tampilan Halaman Input Foto	Mengakses tampilan halaman input foto	Halaman dapat menampilkan form input foto
		Fungsional Tampilan Halaman Input Foto	Mengakses Tampilan halaman input foto	Data dapat diisi dan dapat disimpan
		Fungsional Tampilan Halaman Simpan Foto	Mengakses Tampilan halaman simpan foto	Halaman dapat menyimpan foto
		Fungsional Tampilan Halaman	Mengakses tampilan halaman	Halaman dapat menampilkan data foto tersimpan
		Lihat Foto	lihat foto	
4	Menu Input Cuti	Fungsional Tampilan Halaman Input Cuti	Mengakses tampilan halaman input cuti	Halaman dapat menampilkan form input cuti
		Fungsional Tampilan Halaman Input Cuti	Mengakses tampilan halaman input cuti	Data dapat diisi dan dapat disimpan

Tabel 3.37 Daftar Pengujian pada Pegawai (Lanjutan)

<b>NO</b>	<b>KELAS UJI</b>	<b>DAFTAR PENGUJIAN</b>	<b>KASUS UJI</b>	<b>HASIL YANG DIHARAPKAN</b>
4	Menu Input Cuti	Fungsional Tampilan Halaman Simpan Cuti	Mengakses tampilan halaman simpan cuti	Halaman dapat menyimpan data cuti
		Fungsional Tampilan Halaman Lihat Cuti	Mengakses tampilan halaman lihat cuti	Halaman dapat menampilkan data cuti tersimpan
5	Menu Input Pensiun	Fungsional Tampilan Halaman Input Pensiun	Mengakses Tampilan Halaman input pensiun	Halaman dapat menampilkan form input pensiun
		Fungsional Tampilan Halaman Input Pensiun	Mengakses Tampilan Halaman input pensiun	Data dapat diisi dan dapat disimpan
		Fungsional Tampilan Halaman Simpan Pensiun	Mengakses Tampilan Halaman Simpan Pensiun	Halaman dapat menyimpan data pensiun
		Fungsional Tampilan Halaman Lihat Pensiun	Mengakses Tampilan Halaman lihat pensiun	Halaman dapat menampilkan data pensiun tersimpan

Tabel 3.37 Daftar Pengujian pada Pegawai (Lanjutan)

<b>NO</b>	<b>KELAS UJI</b>	<b>DAFTAR PENGUJIAN</b>	<b>KASUS UJI</b>	<b>HASIL YANG DIHARAPKAN</b>
6	Menu Input Mutasi	Fungsional Tampilan Halaman  Input Mutasi	Mengakses tampilan halaman  input mutasi	Halaman dapat menampilkan form input mutasi
		Fungsional Tampilan Halaman Input Mutasi	Mengakses tampilan halaman input mutasi	Data dapat diisi dan dapat disimpan
		Fungsional Tampilan Halaman Simpan Mutasi	Mengakses Tampilan halaman simpan mutasi	Halaman dapat menyimpan data mutasi
		Fungsional Tampilan Halaman Lihat Mutasi	Mengakses tampilan halaman lihat mutasi	Halaman dapat menampilkan data mutasi tersimpan

Tabel 3.38 Daftar Pengujian pada Kadis

<b>NO</b>	<b>KELAS UJI</b>	<b>DAFTAR PENGUJIAN</b>	<b>KASUS UJI</b>	<b>HASIL YANG DIHARAPKAN</b>
1	Menu Verifikasi Cuti	Fungsional Tampilan Halaman Verifikasi Cuti	Mengakses tampilan halaman verifikasi cuti	Halaman dapat menampilkan data cuti yang telah diajukan oleh pegawai
		Fungsional Tampilan Halaman Lihat Pengajuan Cuti	Mengakses tampilan halaman lihat pengajuan cuti	Halaman dapat menampilkan data cuti yang diajukan
		Fungsional Tampilan Halaman Acc Pengajuan Cuti	Mengakses tampilan halaman acc pengajuan cuti	Halaman dapat menampilkan hasil pengajuan cuti yang diajukan
2	Menu Verifikasi Mutasi	Fungsional Tampilan Halaman Verifikasi Mutasi	Mengakses tampilan halaman verifikasi mutasi	Halaman dapat menampilkan data mutasi yang diajukan oleh Pegawai
		Fungsional Tampilan Halaman Lihat Pengajuan Mutasi	Mengakses tampilan halaman lihat pengajuan mutasi	Halaman dapat menampilkan data mutasi yang diajukan
		Fungsional Tampilan Halaman Acc Pengajuan Mutasi	Mengakses tampilan halaman acc pengajuan mutasi	Halaman dapat menampilkan hasil pengajuan mutasi yang diajukan

Tabel 3.38. Daftar Pengujian pada Kadis (Lanjutan)

<b>NO</b>	<b>KELAS UJI</b>	<b>DAFTAR PENGUJIAN</b>	<b>KASUS UJI</b>	<b>HASIL YANG DIHARAPKAN</b>
3	Menu Cetak Laporan	Fungsional Tampilan Halaman Cetak	Mengakses tampilan halaman cetak laporan	Menampilkan halaman cetak
		Fungsional Tampilan Halaman Pilih Data	Mengakses tampilan halaman pilih data	Halaman dapat menampilkan data terpilih
		Fungsional Tampilan Halaman Print atau Cetak	Mengakses tampilan halaman print atau cetak	Halaman dapat menampilkan hasil akhir

## V. KESIMPULAN DAN SARAN

### 5.1. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan dapat disimpulkan sebagai berikut.

1. Telah berhasil dibangun Sistem Informasi Kepegawaian Dinas Pemuda dan Olahraga (DISPORA) Provinsi DKI Jakarta berbasis *Web* menggunakan *Framework* Laravel yang dapat digunakan untuk penyimpanan data kepegawaian serta meningkatkan kinerja pegawai.
2. Sistem informasi ini dapat membantu dinas dalam melakukan penyusunan pendataan pegawai dengan adanya fitur *generate* dari data sebelumnya dengan syarat data kepegawaian tersebut telah ada pada sistem sehingga *admin* hanya perlu memperbaharui data atau langsung menyimpan data. Apabila data pegawai belum ada pada sistem, maka fitur *generate* data tidak dapat berfungsi, tetapi *field* pendataan data pegawai telah diberikan *value* awal sehingga *admin* hanya perlu memperbaharui atau menambahkan masukkan dalam proses pendataan pegawai dan *admin* tidak perlu untuk mengatur format *output* dalam proses pendataan tersebut.
3. Sistem informasi ini dapat membantu melakukan pencarian data pegawai yang telah dimasukkan oleh *admin*, data pegawai dapat dikelompokkan berdasarkan golongan atau pangkat dan ditampilkan dalam bentuk tabel dan *tab*

*view* berdasarkan jenis nip, perangkat maupun golongan sehingga dapat dengan mudah ditemukan apabila dokumen tersebut dibutuhkan pada kemudian hari.

## 5.2. Saran

Berdasarkan perancangan dan hasil implementasi sistem yang dilakukan, maka beberapa saran yang perlu diperhatikan dalam mengembangkan sistem ini adalah sebagai berikut.

1. Menambahkan fitur grafik data yang menampilkan jumlah data pada masing-masing *menu* secara *realtime* berdasarkan pada hak akses *admin* serta kadis.
2. Menambahkan fitur pengumuman pada pegawai dan *notifikasi* pada *admin* dan kadis sehingga memudahkan dalam menyampaikan informasi.
3. Menambahkan fitur prediksi kata dalam proses pendataan data kepegawaian.
4. Mengembangkan sistem agar dapat menyimpan data dalam kapasitas besar untuk dinas maupun pada cabang lainnya dalam pendataan pegawai.

## DAFTAR PUSTAKA

Aminudin. 2015. *Cara Efektif Belajar Framework Laravel*. Edited by Lukmanul Hakim. Yogyakarta: Lokomedia.

Awaludin, R. (2016). *Seminggu Belajar Laravel*. Seri Seminggu Belajar Laravel, 10.

Februariyanti, dan Zuliarso. 2012. *Rancang Bangun Sistem Perpustakaan Untuk Jurnal Elektronik*. Jurnal Teknologi Informasi Dinamika Vol 17 No 2.

Gregorius. 2000. *Membuat Homepage Interaktif*. Jakarta: PT. Elex Media Koputindo.

[Http://dispora.jakarta.go.id](http://dispora.jakarta.go.id) diakses pada tanggal 20 Januari 2019, 09.00 WIB.

Jiang, F., & Lu, Y. (2012). *Software testing model selection research based on Yin-*

*Yang testing theory*. *IEEE Proceeding of International Conference on Computer*, 590-594.

Jogiyanto, H. M. 2001. *Analisis dan Desain Sistem Informasi*. Yogyakarta: Andi Offset.

Kristanto. 2002. *Perancangan Sistem Informasi dan Aplikasinya*. 1th ed, Yogyakarta : Gava Media.

Madcoms. 2011. *Aplikasi Web Database dengan Dreamweaver dan PHP- MySQL*. Andi Offset. Yogyakarta.

Mall, R. 2003. *Fundamentals of Software Engineering*. Prentice - Hall of India Private Limited. New Delhi.

Munawar. 2005. *Pemodelan Visual Dengan UML*. Graha Ilmu. Yogyakarta.

Pitt, Chris. 2012. *Pro PHP MVC*. New York: Apress.

Pressman, R. S. (2010). *Software Engineering A Practitioners Approach Seventh Edition*. New York: McGraw-Hill Companies, Inc.

- Ramadhan, Nugroho. 2009. *Desain Web dengan PHP*. STMIK Triguna Dharma. Medan.
- Robi'in, B. 2002. *Mengolah Database dengan SQL pada Interbase menggunakan Delphi 6.0*. Yogyakarta: Andi Offset.
- Rohman, a. (2014). best PHP frameworks for 2014. Mengenal Framework "Laravel" (Best PHP Frameworks For 2014).
- Simarmata, J. 2007. *Perancangan Basis Data*. Yogyakarta: Andi Offset.
- Soedaryono. 2000. *Tata Laksana Kantor*. Jakarta: Bumi Aksara.
- Susanto, A. 2004. *Sistem Informasi Manajemen Konsep dan Pengembangannya*. Lingga Jaya. Bandung.
- Sutarman. 2012. *Pengantar Teknologi Informasi*. Jakarta: PT. Bumi Aksara.
- Timothy Lethbridge dan Robert Laganier. 2002:151. *Object Oriented Software Engineering: Pratical Software Development Using UML and Java*. Mc Graw Hill-Education. UK.
- Triyono, Liliek. 2016. *Sistem Informasi Akademik Kampus Berbasis Web dengan LARAVEL 5*. Yogyakarta: Lokomedia.
- Tulangow, B.M. 2011. "Sistem ujian berbasis web". *Jurnal teknomatika*, Volume 1, No. 1, Januari 2011. STMIK PalComTech. Palembang.
- Whitten, Jeffery, L 2004:430. *Metode Desain dan Analisis Sistem*. Yogyakarta: Andi Offset.
- Yuhefizer. 2003. *Tutorial Ilmu Komputer dan Jaringan*. <http://IlmuKomputer.com>
- /Tutorial-Komputer-dan-Jaringan.doc, diakses pada tanggal 01 September 2018, Pukul 20.00 WIB.